HTL Mistelbach

Höhere Lehranstalt für

Biomedizin- und Gesundheitstechnik

DIPLOMARBEIT

Planify App

Ausgeführt im Schuljahr 2013/14 von:

Martin Thoma 5AHELB-24

Martin Weise 5BHELB-25

Betreuer/Betreuerin:

DI Paul Tutschek

Mistelbach, am 12.05.2014

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche erkenntlich gemacht habe.

Mistelbach, am 12.05.2014

Martin Thoma: .....

Martin Weise: .....

Danksagung

Wir bedanken uns bei:

- DI Paul Tutschek, unserem Betreuer, dafür, dass er uns dieses Projekt vorgeschlagen hat, uns bei der Verwirklichung des Projekts unterstützt hat und fachlich sehr kompetent beraten hat.

- Christian Heiss, unserem Partner der Heiss & Süß GmbH, für die gute Zusammenarbeit.

- Yves Schulz, vom Sektor 5, der uns steuerrechtlich beraten hat und uns Kontakte sowie wichtige Tipps für die Zukunft mitgegeben hat.

- Mag. Reinhard Gabauer, unserem Wirtschaft & Recht – Professor, der uns ebenfalls steuerrechtlich beraten hat.

- Philip Ehrenfellner und Christoph Peter von der Internetkultur, die uns mit Insiderwissen versorgt haben.

- DI Martin Panzer, der uns bei der Kundenbetreuung beraten hat.

Kurzfassung

Das Ziel dieser Diplomarbeit war es eine Anwendung (Software) für unseren Partner (Heiss & Süß GmbH) zu entwickeln. Die Anwendung kann auf allen Plattformen verwendet werden, die einen modernen Webbrowser installiert haben. Mithilfe dieser Anwendung kann unser Partner das Unternehmen verwalten, seine Mitarbeiter effizienter einsetzen und den Produktionszyklus besser überwachen. Für jede Abteilung kann ein Tagesplan mit allen Arbeitsschritten ausgedruckt werden, auf dem der verantwortliche Mitarbeiter unterschreiben muss, damit nachvollziehbar ist, wer - wann - was produziert hat. Für den Anwender besteht die Anwendung im Prinzip aus fünf Teilen:

- Unter Einstellungen werden alle Mitarbeiter, Produkte und Lieferanten definiert.
- Im Dienstplan wird definiert, welche Mitarbeiter zur Verfügung stehen.
- Unter Produktion können dann die (von dem Anwender / der Anwenderin) definierten Produkte in Form einer Bestellung aufgegeben werden.
- Im Wochenplan können diese Aufträge an bestimmten Tagen eingeteilt werden.
- Diese erscheinen dann im Tagesplan detailliert, wo Arbeitsschritte und Mitarbeiter zwischen Abteilungen verschoben werden können. Dieser Tagesplan ist anschließend ausgedruckbar.

Im Zuge dieser Diplomarbeit beschäftigen wir uns auch mit der Weiterentwicklung und der Vermarktung der Software.

Abstract

The aim of this diploma thesis was to develop an application (software) for our partner, Heiss & Süß GmbH. The software can be used on all platforms with a modern web browser. With this software-application our partner is able to manage his business. He uses this software to fit in his employees more efficient and to monitor the production cycle. For each section, a plan can be printed for every day with every work step, where the responsible employee has to sign. This makes it possible to trace - who has produced - which product - at what time. The software basically consists of five parts. Under settings all employees, products and suppliers can be defined. In the work plan, a list of all available employees is shown. Under production, the predefined products can be commissioned. In the week plan, these products can be assigned to days. In the detailed day plan the work steps and employees can be displaced between the divisions. This plan can be printed then. In this thesis, we also deal with the development and marketing of our software.

Vorwort

Der Titel unserer Diplomarbeit ist „Planify App“, weil sie sich vorrangig mit der Planung eines Unternehmens mit Software beschäftigt, welche vorrangig auf mobilen Geräten verwendet werden soll. Wir haben uns schon früh für ein gemeinsames, softwarelastiges Projekt als Diplomarbeit entschieden, weil wir uns sehr für die Programmierung und Planung von bestimmten Abläufen im Sektor „Business Software“ interessieren. Als wir den Projektvorschlag von unserem Softwaredesign Professor DI Paul Tutschek erhielten, haben wir uns sofort dafür entschieden, da die Zusammenarbeit mit einem Kunden für uns eine neue Herausforderung darstellte. Wenn man ein Projekt für einen Kunden durchführt lernt man viel mehr dabei, da auf der Ergebnisseite jemand steht, der sich ein bestimmtes Ergebnis des Projektes erwartet, was in den Abschnitten „Entstehungsgeschichte“ und „Gelerntes“ noch näher ausgeführt wird.

Einleitung

Das Ziel dieses Projektes ist es den Planungsablauf unseres Kunden in eine Web-Anwendung zu integrieren.

Die Web-Anwendung soll einfach zu bedienen sein, den Geschäftsprozess nachvollziehbar darstellen und die Rückverfolgbarkeit der Produktion soll gegeben sein, indem ein verantwortlicher Mitarbeiter auf einem ausgedruckten Arbeitsplan jeden Arbeitsschritt unterzeichnen muss. Genauere Informationen über die Anforderungen des Projekts finden sich im Anhang unter „Pflichtenheft“ und „Werkvertrag“.

Die Benutzeroberfläche wird anschließend unter „Bedienung“ näher erklärt.

Unter „Programmieren (Theorie)“ und „Programmieren (Umsetzung)“ werden Software-Aspekte näher behandelt. Diese Web-Anwendung ist in HTML, CSS und JavaScript geschrieben worden. Die Daten werden in einer MySQL-Datenbank gespeichert, welche über PHP manipuliert wird.

Literaturverzeichnis

[Sei06] Seidler, Kai; Vogelsang, Kay: *Das XAMPP-Handbuch: Der offizielle Leitfaden zu Einsatz und Programmierung.* Pearson-Wesley Verlag, München 2006.

[Zil13] Zillgens, Christoph, Gangelt: Responsive Webdesign: Reaktionsfähige Websites gestalten und umsetzen. Carl Hanser Verlag, München 2013.

[Mau12] Mauriece, Florence: *PHP 5.4 & MySQL 5.5: Der Einstieg in die Programmierung dynamischer Websites.* Addison-Wesley Verlag, München 2012.

[Ste07] Steyer, Ralph: *JavaScript. Einstieg für Anspruchsvolle.*Pearson-Wesley Verlag, München 2007

[Fek04] Fekete, Thomas: Customer Relationship Management im Wandel der Zeit mit spezieller Berücksichtigung des Electronic Customer Relationship Management der österreichischen Wirtschaft, Wien 2004.

[Tan07] Tanoury, Doug: Back in Black: Understanding The Financial Impact of CRM: Online in Internet: URL: http://www.crm2day.com/library/EpulEyyFpkkBEVfRsg.php ;

[Mai07] Maida, Martina; Wickenhauser, Sandra: *Seminararbeit: Thema: Customer Relationship Management*, Wirtschaftsuniversität, Wien 2007.

[Wal04] Waldemar Pelz; Strategisches und operatives Management, Ein Leitfaden zur Erstellung eines professionellen Marketing-Plans, Norderstedt 2004

Inhaltsverzeichnis

Eidesstattliche Erklärung 2
Danksagung 3
Kurzfassung 4
Abstract 4
Vorwort 5
Einleitung 5
Literaturverzeichnis 6
Inhaltsverzeichnis 7
Abbildungsverzeichnis 11
Tabellenverzeichnis 12
Abbildungsnachweis 12
Abkürzungsverzeichnis 13
1. Projektteam 14
2. Zeiterfassung 15
3. Projektplanung 16
3.1. Meilensteine 16
3.1. Kosten 16
4. Markt- & Konkurrenzanalyse 17
4.1. SWOT-Analyse(in Anlehnung an [Wal04]) 17
4.1.1. Unternehmensanalyse (interne Analyse) 17
4.1.2. Umfeldanalyse (externe Analyse) 18
4.1.3. Kombination 18
4.1.4. Bewertung 20
4.1.5. Unternehmensstrategie definieren 20
4.2. Ziele 20
4.3. Vorteile 20
4.4. Nachteile 20
5. Kundenmanagement 21
5.1. Grundlagen (Theorie) 21
5.1.1. Ziele 21
5.1.2. Sekundärziele (in Anlehnung an [Mai07]) 22
5.1.3. Aufbau 24

5.2. Erkenntnisse 25
6. Entstehungsgeschichte 26
7. Gelerntes 28
7.1. Events 28
7.1.1. Überall-App Kongress 28
7.1.2. Kod.io Linz 28
8. Ausblick 29
9. Bedienung 30
9.1. Login, Menü 30
9.2. Dienstplan 30
9.2.1. Leiharbeiter hinzufügen 31
9.3. Wochenplan 31
9.4. Tagesplan 32
9.5. Produktion 33
9.5.1. Spezialbestellung 33
9.6. Einstellungen 34
9.6.1. Produkte 34
9.6.2. Produktgruppen 35
9.6.3. Mitarbeiter 35
9.6.4. Lieferanten 36
9.6.5. Posten 37
9.7. Konten 37
10. Programmieren (Theorie) 39
10.1. Allgemeines 39
10.1.1. Prinzip 39
10.2. Sprachen 39
10.2.1. HTML 39
10.2.2. CSS 44
10.2.3. JavaScript 47
10.2.4. PHP 53
10.2.5. MySQL 57
10.2.6. XAMPP 59
10.2.7. Workflow (in Anlehnung an [Zil13]) 62

11. Programmieren (Umsetzung) 65
11.1. Entwicklungsumgebung 65
11.1.1. Funktionen 65
11.2. Datei-Struktur 66
11.2.1. ajax-Verzeichnis 66
11.2.2. cntnt-Verzeichnis 66
11.2.3. css-Verzeichnis 66
11.2.4. font-Verzeichnis 66
11.2.5. fpdf-Verzeichnis 67
11.2.6. img-Verzeichnis 67
11.2.7. js-Verzeichnis 67
11.2.8. sql-Verzeichnis 67
11.2.9. var-Verzeichnis 67
11.3. Front-End 68
11.3.1. Template 68
11.3.2. main.js-Datei 72
11.3.3. Komponenten 73
11.4. Back-End 82
11.4.1. Template 82
11.4.2. SQL-Klasse 86
11.4.3. Komponenten 87
Anhang 104
I. Pflichtenheft 104
Allgemeines 104
Einleitung 105
Charakteristika 105
Lieferumfang 106
Einflussfaktoren 106
Farbliche Hervorhebungen: 106
Einstellungen 108
Produkte 108
Mitarbeiter 108
Lieferanten 108

Posten....108
Risiken... 109
Verpflichtungen 109
II. Werkvertrag 109
III. Ablaufdiagramme 110
IV. Zeiterfassung 122
V. DVD 125

Abbildungsverzeichnis

Abb. 1: Diagramm Zeitverteilung 15
Abb. 2: SWOT-Matrix 19
Abb. 3: Infografik zum Kundenmanagement....21
Abb. 4: Musterbeispiel für gute Kundenintegration 23
Abb. 5: Erste Skizzierungen beim Kundentreffen 26
Abb. 6: v.l.n.r.: Martin Weise, Martin Thoma vor dem Ars Electronica Center in Linz....29
Abb. 7: Planify Login 30
Abb. 8: Planify Dienstplan 31
Abb. 9: Planify Leiharbeiter 31
Abb. 10: Planify Wochenplan 32
Abb. 11: Planify Tagesplan....33
Abb. 12: Planify Produktion 33
Abb. 13: Planify Spezialbestellung 34
Abb. 14: Planify Einstellungen Produkte....34
Abb. 15: Planify Einstellungen Neues Produkt Initialansicht 35
Abb. 16: Planify Einstellungen Produktgruppen....35
Abb. 17: Planify Einstellungen Mitarbeiter 36
Abb. 18: Planify Einstellungen Leiharbeiterfirmen 36
Abb. 19: Planify Einstellungen Lieferanten 37
Abb. 20: Planify Einstellungen Posten....37
Abb. 21: Planify Einstellungen Konten....38
Abb. 22: Zugriff auf Hardware mit HTML5 42
Abb. 23: Formular in HTML5 mit placeholder-Attributen 43
Abb. 24: Tastaturen bei verschiedenen Eingabetypen....44
Abb. 25: Bootstrap Logo....47
Abb. 26: Das Document Object Model (DOM) 48
Abb. 27: Beispielanwendung unter Verwendung von JavaScript 49
Abb. 28: Ablauf der Aufgabenstellung 51
Abb. 29: Flussdiagramme HTTP-Request (li.) & kombiniert mit AJAX-Request (re.) 52
Abb. 30: Request-Response Modell mit PHP-Parser....54
Abb. 31: Formularverarbeitung mit PHP 57
Abb. 32: Tabelle pf_colors....58
Abb. 33: Anteil der Webserver-Technologien 61
Abb. 34: Accessability von Elementen bei Touch-Oberflächen....63
Abb. 35: Gelungenes Design über verschiedene Bildschirmauflösungen 64
Abb. 36: PhpStorm 7.0 Entwicklungsumgebung....65
Abb. 37: Wochenplan....76
Abb. 38: Dienstplan....77
Abb. 39: Tagesplan....80
Abb. 40: Beispielabfrage aus Code 43. 91
Abb. 41: Flussdiagramm Dienstplan....110
Abb. 42: Flussdiagramm Einstellungen 111

Abb. 43: Flussdiagramm Firmen....111
Abb. 44: Flussdiagramm Konten....112
Abb. 45: Flussdiagramm Leiharbeiter....112
Abb. 46: Flussdiagramm Lieferanten....113
Abb. 47: Flussdiagramm Login....113
Abb. 48: Flussdiagramm Menü....114
Abb. 49: Flussdiagramm Mitarbeiter....115
Abb. 50: Flussdiagramme Posten....116
Abb. 51: Flussdiagramm Produkt....117
Abb. 52: Flussdiagramm Produkte....118
Abb. 53: Flussdiagramm Produktgruppen....119
Abb. 54: Flussdiagramm Produktion....119
Abb. 55: Flussdiagramm Spezialbestellung....120
Abb. 56: Flussdiagramm Tagesplan....120
Abb. 57: Flussdiagramm Wochenplan....121

Tabellenverzeichnis

Tabelle 1: Tabellarische Zeiterfassung....15
Tabelle 2: Meilensteine....16
Tabelle 3: Kostenaufschlüsselung....16
Tabelle 4: Pflichtenheft Kontaktinformationen....104
Tabelle 5: Pflichtenheft Projektdaten....104
Tabelle 6: Pflichtenheft Projektcharakteristika....105
Tabelle 7: Pflichtenheft Lieferumfang....106
Tabelle 8: Pflichtenheft Risiken....109
Tabelle 9: Zeiterfassung Detailaufschlüsselung....124

Abbildungsnachweis

Abb. 2 http://www.lernende-regionen.at/de/page.asp?id=30 (Stand 28.04.2014)

Abb. 23 http://failconference.at (Stand: 16.02.2014)

Abb. 25 http://getbootstrap.com/about/ (Stand: 16.02.2014)

Abb. 33 http://www.netcraft.com/survey/ (Stand: 08.02.2014)

Abb. 3 Blaue Grafiken von http://iconarchive.com (Stand: 05.05.2014)

Anmerkung:

Nicht angeführte Bilder sind Eigenaufnahmen.

Abkürzungsverzeichnis

Name	Bedeutung
HTML	Hyper Text Markup Language
W3C	World Wide Web Consortium
Syntax	„Programmier-Grammatik“
Tag	HTML-Attribut
CSS	Cascading Stylesheet
Cursor	Mauszeiger
AJAX	Asynchronous JavaScript And XML
JSON	JavaScript Object Notation
PHP	Hypertext Preprocessor
SQL	Structured Query Language
Patch	Programm, um Software zu erneuern
root	Administrator unter Linux-Platform
960px-Grid	Bekanntes Skaliermodell (Webentwicklung)
IDE	Integrated Development Environment
FPDF	PDF Bibliothek für PHP
MVC	Model View Controller (Webentwicklung)
iOS	Betriebsystem für Mobilgeräte
Slider	Schieberegler
Login	Anmeldeverifizierung
Logout	Verlassen eines Dienstes
DSN	Data Source Name
EXIF	Exchangeable Image File Format
JPEG	Bildkompressionsformat

1. Projektteam

Martin Thoma

5AHELB – HTL Mistelbach

thomamartin@gmx.at

Simonsfeld 7

2115 Ernstbrunn

Martin Weise

5AHELB – HTL Mistelbach

hi@martinweise.at

Försterweg 29

2136 Laa an der Thaya

2. Zeiterfassung

Name	Zeit (in Stunden)
Martin Thoma	207,5
Martin Weise	240,5
Zeit Gesamt	448

Tabelle 1: Tabellarische Zeiterfassung

Zeitverteilung:

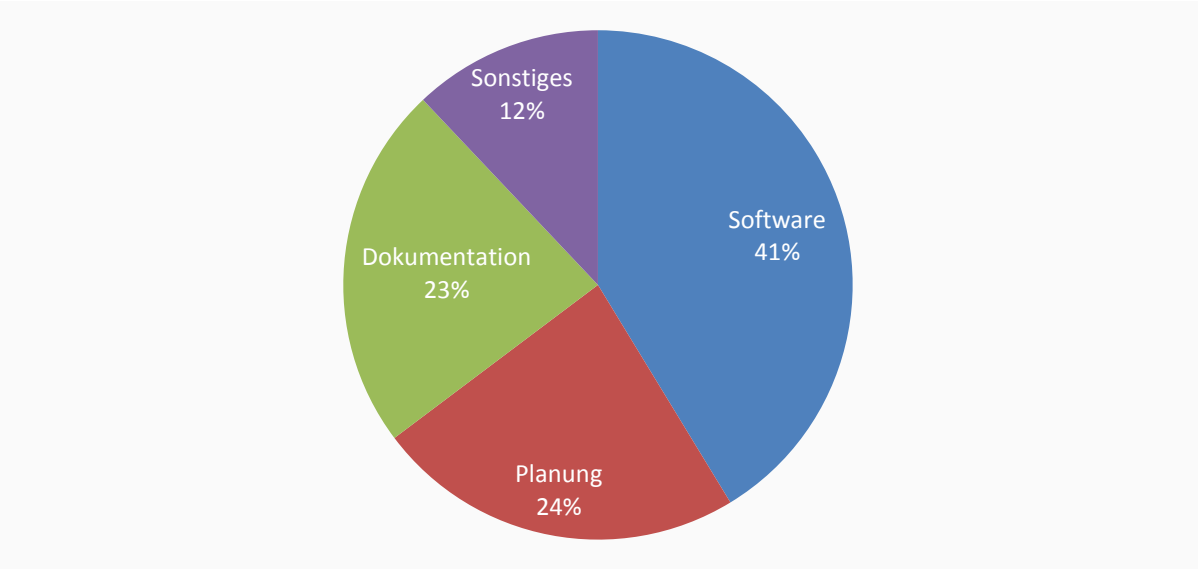

Abb. 1: Diagramm Zeitverteilung

Wie in Abb. 1 zu sehen ist, wurde die meiste Zeit in die Entwicklung der Software gesteckt. Das hat zum Teil damit zu tun, dass die Planung der Software im Vorhinein schlecht realisierbar war, zum Teil aber natürlich damit, dass die Planung mangelhaft in der Ausführungsqualität war.

Würden wir vor ein vergleichbares Projekt gestellt werden und müssten dieses von der Planung bis zur Abgabe realisieren, wären nur die Hälfte der Software-Zeit von nöten, weil wir bei der Erstellung dieser Software gelernt haben.

Wir würden unser Hauptaugenmerk auf komplett andere Dinge legen und würden anders an den Kunden herantreten, sodass dieser kompetenter Beraten wird und uns durch qualitativ bessere Aussagen besser unterstützen würde.

3. Projektplanung

3.1. Meilensteine

Ist	Soll	Beschreibung
16.05.2013	01.06.2013	Pflichtenheft unterschrieben
02.04.2013	01.07.2013	Design-/Konzeptentwurf präsentieren
18.08.2013	01.09.2013	Prototyp fertigstellen
13.10.2013	13.10.2013	Beginn der Testphase und Bedienungsunterweisung
28.10.2013	01.11.2013	Fertigstellung
21.10.2013	15.12.2013	Bedienungsanleitung fertig
10.02.2013	15.01.2014	Endgültige Abnahme und Übergabe der App
21.02.2013	01.03.2014	Theoretischer Teil der Diplomarbeit fertig
22.04.2013	15.04.2014	Diplomarbeit fertig
12.05.2014	20.05.2014	Abgabe Diplomarbeit

Tabelle 2: Meilensteine

Mit der endgültigen Abnahme des Projektes gab es Verzögerungen, da anfangs der Server unseres Kunden noch nicht vorhanden bzw. noch nicht in Betrieb war und als dieser dann funktionierte war es schwierig einen Termin zu vereinbaren, an dem alle Beteiligten Zeit hatten.

Die Diplomarbeit wurde erst eine Woche später als geplant fertiggestellt, da die Schule neue Anforderungen an die Diplomarbeit stellte und wir unsere Diplomarbeit dahingehend noch anpassen mussten.

3.1. Kosten

Name	Aufwand (in Euro)
PHPStorm7 (2 Stück)	180
Photoshop Elements 12	95
Visio Professional (60T Probeversion)	0
Diplomarbeit drucken (5 Stück)	140
Verkauf App	2000
USt (20%)	0
Erlös	1585

Tabelle 3: Kostenaufschlüsselung

4. Markt- & Konkurrenzanalyse

Während unseres Projektes haben wir uns oft überlegt, welche Stärken und Schwächen unser Projekt und auch welche Chancen man mit diesem Projekt am Markt hat und wie es mit der Konkurrenz aussieht. Im Wesentlichen haben wir dabei nichts anderes gemacht, als eine SWOT-Analyse.

4.1. SWOT-Analyse (in Anlehnung an [Wal04])

Die SWOT-Analyse ist ein Hilfsmittel des strategischen Managements. Sie ist auch die Grundlage vieler Marketingstrategien. „SWOT“ steht für „Strengths”, „Weaknesses”, „Opportunities” und „Threats”. Also ist die SWOT-Analyse eine Analyse von Stärken, Schwächen, Chancen und Risiken. Die internen Stärken und Schwächen werden externen Chancen und Risiken gegenübergestellt. Das Ziel ist es die Unternehmensstrategie zu analysieren. Dazu gehört nicht nur das analysieren, sondern auch die folgenden Veränderungen und Konsequenzen, die aus dieser Analyse hervorgehen.

Die Analyse kann in mehreren verschiedenen Unternehmensbereichen eingesetzt werden, im Zusammenhang mit der Produktpolitik, dem Absatzpotential oder der Standortanalyse.

Die SWOT-Analyse besteht aus 5 Schritten:

- Unternehmensanalyse (interne Analyse)
- Umfeldanalyse (externe Analyse)
- Kombination
- Bewertung
- Definition der Unternehmensstrategie

4.1.1. Unternehmensanalyse (interne Analyse)

Die Unternehmensanalyse wird auch interne Analyse oder Stärken-Schwächen-Analyse genannt. Hierbei werden die internen Stärken und Schwächen eines Unternehmens erkannt und aufgelistet. Es werden das Mitarbeiterteam, die Produkte und die Produktion dieser betrachtet. Die Schwächen und Stärken eines Unternehmens resultieren aus Fähigkeiten und Ressourcen eines Unternehmens. Das Ziel dieser Analyse ist es Ansatzpunkte für die Schaffung eines strategischen Wettbewerbsvorteils zu finden. Wenn man die Stärken seines Unternehmens kennt, kann man auf diesen Stärken aufbauen und diese Stärken auch ausnutzen. Die Schwächen wirken sich negativ auf das Unternehmen aus und hindern es daran einen Wettbewerbsvorteil zu erlangen. Kennt man seine Schwächen kann man diesen Schwächen entgegenwirken. Die Stärken und Schwächen werden verglichen mit den jeweiligen Schwächen bzw. Stärken von Mitbewerbern, da Stärken und Schwächen immer relativ zu betrachten sind. Ein Unternehmen kann einen Aspekt der Analyse als Schwäche sehen, hingegen kann ein anderes Unternehmen

dasselbe Ergebnis als eine Stärke ansehen. Die interne Analyse kann auf mehrere Arten strukturiert werden.

4.1.2. Umfeldanalyse (externe Analyse)

In der Umfeldanalyse oder auch externen Analyse werden die Chancen und Risiken eines Unternehmens identifiziert. Bei dieser Analyse fällt der Blick auf die Aufgaben des Unternehmens, auf den Wettbewerb und auf den potenziellen Markt, also auf die Kunden. Hier gibt es eine Unterteilung in Technologiefrüherkennung & Technologieprognose und Konkurrentenanalyse.

Für die Technologiefrüherkennung und Technologieprognose muss man Signale aus der Unternehmensumfeld aufnehmen und interpretieren. Unter Unternehmensumfeld versteht man die Kunden, also den potenziellen Markt, die Zulieferer des Unternehmens und Institutionen, die Erkenntnisse über neue Technologien liefern können. Weiters können hier die Weiterentwicklungspotenziale neuer Technologien und die Grenzen bekannter Technologien in Erfahrung gebracht werden.

Bei der Konkurrentenanalyse werden die Aktivitäten von relevanten Mitbewerbern in dem Sektor des eigenen Unternehmens untersucht. Zuerst werden die Konkurrenten oder Wettbewerber ausgewählt. Anschließend werden die Beobachtungsthemen festgelegt. Dann folgt die Informationsgewinnung mit anschließender Speicherung der Informationen. Zum Schluss werden diese Informationen ausgewertet und eventuell mit den Ergebnissen früherer Analysen verglichen.

4.1.3. Kombination

Wenn die Unternehmensanalyse und die Umfeldanalyse abgeschlossen sind, kommt es zur Gegenüberstellung dieser Ergebnisse. Für diese Gegenüberstellung wird eine sogenannte Vier-Felder-Matrix verwendet. In jedes Feld wird einer von vier Teilbereichen mit Strategiemöglichkeiten eingetragen. Die vier Teilbereiche sind Stärken/Chancen, Stärken/Risiken, Schwächen/Chancen & Schwächen/Risiken. Hier werden immer die Ergebnisse einer internen Analyse mit denen einer externen Analyse verglichen. Die Chancen können auch als Möglichkeiten und die Risiken auch als Bedrohungen bezeichnet werden.

SWOT- Analyse		Interne Analyse	
		Stärken (Strengths)	**Schwächen (Weaknesses)**
Externe Analyse	**Chancen** (Opportunities)		
	Gefahren (Threats)		

Abb. 2: SWOT-Matrix

Im ersten Feld der Vier-Felder-Matrix oben werden die internen Stärken mit den externen Chancen oder Möglichkeiten verglichen. Ein Unternehmen sollte mit den vorhandenen Stärken die bestehenden Chancen nutzen, zum Beispiel mit einem bestehenden Produkt einen neuen Markt erschließen.

Im zweiten Feld werden die internen Schwächen den externen Chancen oder Möglichkeiten gegenübergestellt. Hier sollen die eigenen Schwächen beseitigt werden um dann die bestehenden Chancen nützen zu können. Zum Beispiel kann ein neues Produkt entwickelt werden, wo die Schwächen dann nicht mehr vorhanden sind, um den Kundenanforderungen gerecht zu werden.

Im dritten Feld werden die internen Stärken den externen Risiken oder Bedrohungen gegenübergestellt. Hier soll ein Unternehmen mit den eigenen vorhandenen Stärken die Gefahren abwehren, die auf das Unternehmen zukommen können. Zum Beispiel kann das Unternehmen mit der Werbung für ein bestehendes Produkt einen neuen Mitbewerber zurückdrängen.

Im vierten und letzten Feld werden die internen Schwächen mit den externen Risiken oder Bedrohungen verglichen. Hier muss das Unternehmen seine eigenen Schwächen beseitigen um bei drohenden Gefahren bestehen zu können. Zum Beispiel indem Qualitätsmängel am eigenen Produkt behoben werden um einen Mitbewerber zurückzudrängen.

4.1.4. Bewertung

Nach der Kombination der Unternehmensanalyse und der Umfeldanalyse kommt die Bewertung dieser Ergebnisse. Die Ergebnisse der einzelnen vier Strategien werden bewertet und sortiert. Diese Bewertung ist nicht immer gleich, da es auf das persönliche Empfinden der Personen ankommt, die diese Ergebnisse bewerten. Die Ergebnisse können nach Wichtigkeit, Dringlichkeit, vorhandene/fehlende Ressourcen bzw. Kompetenzen, Intensität, Kostenaufwand und Zeitaufwand sortiert werden.

4.1.5. Unternehmensstrategie definieren

Aufgrund der vorangegangen Analysen und Bewertungen kann das Unternehmen nun Strategien und Maßnahmen ableiten.

4.2. Ziele

Die SWOT-Analyse soll vor allem dem Management einen Überblick über das gesamte Unternehmen verschaffen. Sie soll mögliche Wettbewerbsvorteile und den internen Handlungsbedarf aufzeigen. Weiters soll diese Analyse vor möglichen Risiken warnen auf bestehende Chancen, die noch nicht genutzt werden, hinweisen. Das Hauptziel der SWOT-Analyse ist jedoch ganz einfach die Unterstützung bei der Entwicklung einer Unternehmensstrategie.

4.3. Vorteile

Zu den Vorteilen zählt, dass die SWOT-Analyse durch die Vier-Feld-Matrix eine sehr einfache Struktur hat. Diese Analyse ist leicht anwendbar und das mit nur wenig Aufwand. Sie verschafft einen Überblick über das gesamte Unternehmen. Es werden positive und negative Aspekte berücksichtigt und nicht nur eine Seite. Ein weiterer Vorteil ist, dass sich die SWOT-Analyse vielfältig anwenden lässt, zum Beispiel auf das ganze Unternehmen, egal wie groß es auch ist, auf eine einzelne Abteilung oder auf einzelne Projekte.

4.4. Nachteile

Nachteile sind, dass es bei einer Analyse von größeren Unternehmen nur sehr oberflächlich beschrieben werden kann, welche Probleme und Lösungen es gibt. Es gibt keinen Bewertungsmaßstab für die vier Teilbereiche, das heißt sie sind gleichwertig, aber wenn das für ein Unternehmen ein Problem sein sollte, wird es dafür garantiert eine Lösung geben. Wie ich finde ist der größte Nachteil, dass das Ergebnis stark von den an der Analyse beteiligten Personen abhängt.

5. Kundenmanagement

5.1. Grundlagen (Theorie)

Abb. 3: Infografik zum Kundenmanagement

5.1.1. Ziele

Gutes CRM (*engl.Customer Relationship Management*; Kundenmanagement) geht mit der Marketingabteilung eines Unternehmens einher. Es sollte stets gemeinsam durchgeführt werden, dabei sollen die Ziele des CRM ident mit denen des Marketings sein (in Anlehnung an [Fek04]).

- Man will durch Analyse des Kunden und des eigenen Unternehmen eine bessere Kundenzufriedenheit erreichen und die Kauffrequenz des Kunden steigern.

- Ein weiteres Ziel des erfolgreichen CRM ist es, aus interessierten Unternehmen Bestands-kunden zu machen. Das kann durch Bindung der Interessenten durch spezielle Aktionen geschehen oder durch maßgeschneiderte Leistungen für eine neue Kundensparte, die man durch den Einsatz von CRM gefunden hat.

- Je länger die Dauer dieser Beziehung ist, desto weniger kostet es das Unternehmen, weil es die Bedürfnisse des Kunden besser kennenlernt und entsprechend dieser handeln kann. Die damit verbundenen Kosten für die Pflege und Fortführung der Beziehung werden reduziert und das Früherkennen von etwaigen Problemen maximiert.

- Für das Unternehmen sind Bestandskunden ein wirtschaftlicher Vorteil, weil neue Kunden anzulocken mit hohen Kosten verbunden ist. Das können Marketingmaßnahmen oder Angebote sein, die das Unternehmen langfristig wieder aufholen muss, um die Kosten zu decken.

- Bestandskunden tolerieren Fehler und Preisänderungen stärker als Neukunden. Sie wissen, dass das Unternehmen gute Arbeit leistet und sind mit dieser auch zufrieden. Daher sind treue Kunden auch großzügiger, wenn es um die Erhöhung von Tarifen oder Pauschalzahlungen geht, sie schätzen den Nutzen einer langfristigen Beziehung höher ein.

- Viele Dieser Ziele sind jedoch nicht auf direktem Wege zu erreichen, daher gilt es sog. Sekundärziele zu erreichen, bevor es zu einer langfristigen Kundenbindung kommt.

5.1.2. Sekundärziele (in Anlehnung an [Mai07])

5.1.2.1. Kundenzufriedenheit

Die Zufriedenheit eines Kunden richtet sich zentral nach den Erwartungen eines Kunden an die beanspruchten Leistungen des Unternehmens. Diese Anforderungen sind subjektiv und können sich von Kunde zu Kunde unterscheiden, es gilt diese Anforderungen herauszufinden und zu erfüllen.

5.1.2.2. Kundennähe

Die Nähe zu einem Kunden spiegelt sich in der Quantität der Interaktionen oder Geschäftstransaktionen wieder. Je häufiger es zu einer Interaktion zwischen beiden Parteien kommt, desto näher sind sich selbige. Je näher diese Beziehung gepflegt wird, umso besser kann ein Unternehmen auch auf die Bedürfnisse einen Kunden eingehen.

5.1.2.3. Kundentreue

Kommt es wiederholt zu Nach- und Wiederkäufen bei einem Unternehmen, spricht man von Kundentreue. Diese treuen Kunden haben sich im Markt für das Unternehmen entschieden, weil es für das verfügbare Budget unter Abwägung speziellen, branchenspezifischen Aspekten die beste Leistung bietet. Zudem sind zufriedene Kunden auch treue Kunden.

5.1.2.4. Kundenintegration

Die Nähe eines Kunden kann durch Einbinden in unternehmensspezifische Prozesse gesteigert werden. Gut integrierte Kunden erhalten mit gesteigerter Wahrscheinlichkeit ein besseres Ergebnis, als weniger integrierte, weil das Unternehmen verstärkt auf integrierte Kunden eingehen kann.

Wie in Abb. 4 zu sehen ist, führte eine gute Kundenintegration zu einem besseren Ergebnis (subjektiv) für den Kunden. Er sah seine Erwartungen und Anforderungen

Auf der anderen Seite steigen für das Unternehmen meist der Aufwand und die Kosten. Ein Beispiel ist die frühe Integration eines Kunden in Planungsphasen, wie es auch bei unserer Diplomarbeit war. Der Kunde war seit der ersten Stunde voll integriert und hat den Entwicklungsprozess maßgeblich beeinflusst.

5.1.2.5. Kundenbindung

Das Unternehmen welches Leistungen oder Produkte anbietet analysiert die Motive des Kunden, der einen Wiederkauf tätigt, genau und nützt dieses für künftige Kaufabsichten. Es soll eine Bindung, die aus rationalen und emotionalen Aspekten besteht, geschaffen werden.

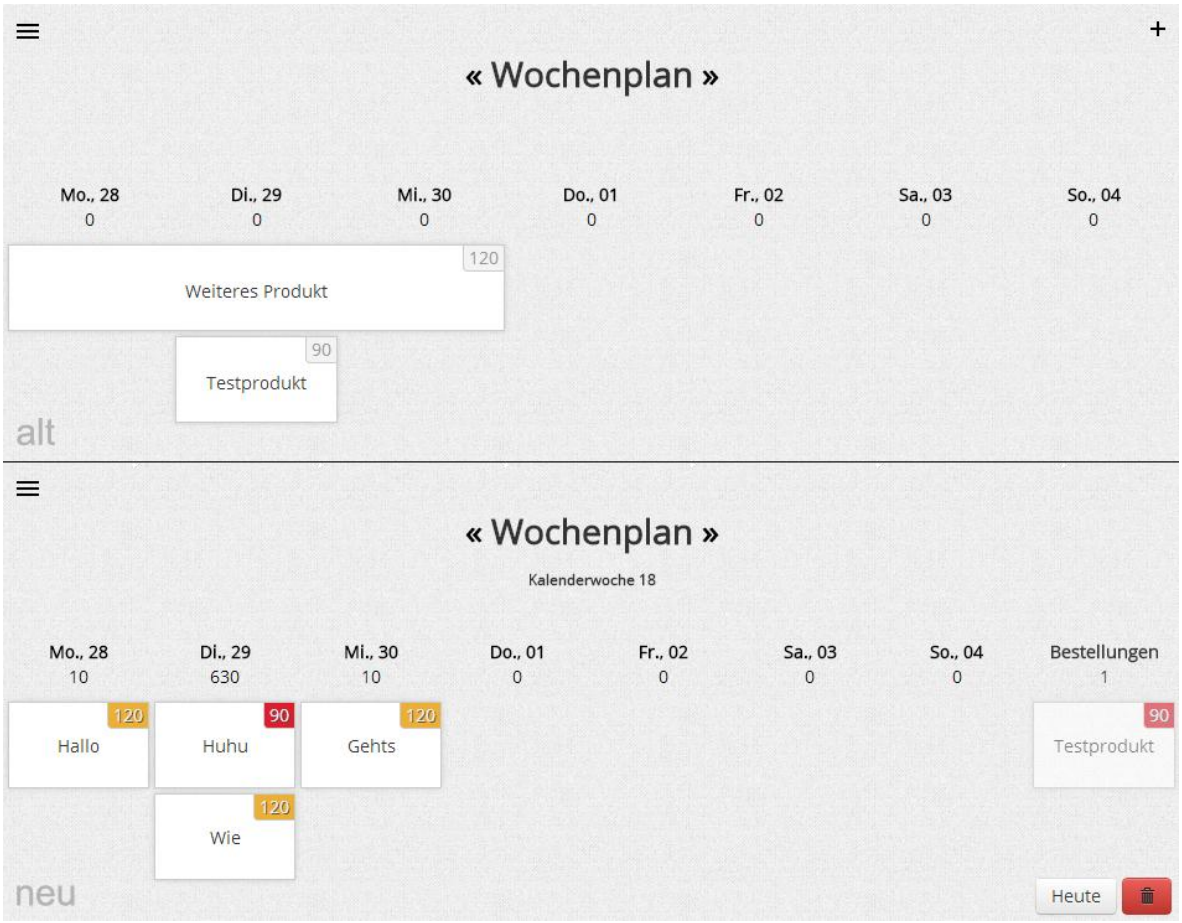

Abb. 4: Musterbeispiel für gute Kundenintegration

5.1.3. Aufbau

5.1.3.1. Strategisches CRM

Man kann das strategische CRM als Teilbereich eines Marketingkonzepts verstehen, der Kunde nimmt dabei eine zentrale Rolle ein, weil sämtliche Bereiche des Unternehmens auf ihn abgestimmt werden. Es wird die Beziehung und das Wechselspiel zwischen Markt und Unternehmen analysiert. Es werden Potenziale erkannt und entsprechende Strategien entwickelt. Die Auswirkungen dieser Strategien betreffen das ganze Unternehmen und sind langfristig orientiert.

5.1.3.2. Analytisches CRM

Die aus einer gründlichen Analyse stammenden Daten sind wichtig für ein CRM-System. Sie müssen gesammelt und ausgewertet werden. Sie dienen als Entscheidungsgrundlage für das Regeln von Kundenbeziehungen. Die Menge der gesammelten Daten ist irrational, wenn die Quantität der Daten ihre Qualität übersteigt und man aus diesen nichts mehr herauslesen kann.

Aktivitäten und Softwarelösungen, die nach dem analytischen Prinzip arbeiten berechnen anhand der Daten diverse Eigenschaften der Kundenbeziehungen und ermöglichen Prognosen für das Unternehmen.

So ist es zum Beispiel möglich aus den Daten Merkmale einer neuen Zielgruppe zu extrahieren oder Eigenschaften, Verhaltensweisen und Wertschöpfungspotenziale von Kunden besser kennenzulernen und einzuschätzen.

5.1.3.3. Operatives CRM

Das operative CRM soll die Ziele des strategischen CRM möglichst so, wie definiert, umsetzen und so Marketing, Service und Vertrieb automatisieren. Die Ergebnisse des analytischen CRM werden im operativen CRM verwendet.

Der Kontakt zum Kunden ist die Basis des Operativen CRM. Diese wird üblicherweise mit einem Produkt gesichert. Die Wichtigkeit des zuständigen Mitarbeiters variiert dabei stark von der Art der Kontaktierung. Wird über Onlinemedien mit dem Kunden interagiert, ist der Mitarbeiter eher nicht entscheidend wichtig. Tritt jedoch Kontakt über direktem Wege ein, wie durch persönliche Treffen und Gespräche, ist der zuständige Mitarbeiter sehr wichtig für das Unternehmen.

Operative CRM-Systeme sollen Mitarbeiter in diesen Schlüsselpositionen im Bereich der Terminplanung, Angebotsverwaltung oder der Verwaltung der Kundendaten unterstützen.

5.1.3.4. Kommunikatives CRM

Beim Kommunikativen CRM ist es wichtig den Kunden nicht mit Widersprüchen oder nicht strategisch geplanten Methoden zu überhäufen. Es sollen alle Kommunikationskanäle, wie

Telefon, Fax, Internet, Email, etc. als gebündelte Einheit agieren. Der Kunde soll gezielt mit einer einheitlichen Zielsetzung angesprochen werden.

Es soll Kontakt zu den Kunden über alle Kanäle gleichzeitig aufgenommen werden, aber niemals beim selben Kunden mehrmals hintereinander.

5.2. Erkenntnisse

Oft ist es leichter mit dem Kunden sofort auf einer etwas lockereren, stets geschäftlichen Basis zu betreuen, weil er sich dann auch mehr sagen traut und einem Fehler besser verzeiht.

Außerdem kann man die Bedienung von Sachen nicht detailliert genug beschreiben, es wird immer wieder Leute geben, welche genau vorgegebenen Schritten einer Anleitung nicht folgen können/möchten.

Alles in allem waren wir äußert froh diese Erfahrungen machen zu dürfen, wir haben sehr viele Zusammenhänge zwischen der Theorie und der Praxis im Arbeitsleben knüpfen können und werden an das nächste Projekt gewiss anders rangehen und wieder neue Erfahrungen machen, die aus uns bestimmt noch kompetentere Absolventen machen.

6. Entstehungsgeschichte

Dieses Projekt beginnt damit, dass unser Projektbetreuer an uns herangetreten ist und uns dieses Projekt in Zusammenarbeit mit einem Kunden bzw. Partner vorgeschlagen hat. Bei unserem Kunden handelt es sich um die Konditorei Heiss & Süß GmbH aus Leitsberg, 3033 Altlengbach. Unser Ansprechpartner seitens der Heiss & Süß GmbH, Christian Heiss, ist zuvor an den Projektbetreuer herangetreten und hat ihn gebeten dieses Projekt mit einer Schülergruppe zu verwirklichen.

Nachdem dieses Projekt Gestalt annahm, setzten wir uns mit Herrn Heiss in Verbindung und vereinbarten ein erstes Treffen. Bei diesem Treffen zeigte er uns, wie derzeit die Arbeitsplanung in seinem Betrieb aussieht und welche Vorstellungen er von dem ganzen Projekt hat. Er verschaffte uns auch Einblicke in die Abläufe im Betrieb, damit wir uns von seinen Vorstellungen ein Bild machen konnten. Dieses erste Treffen fand am 21. Februar 2013 in der HTL Mistelbach statt.

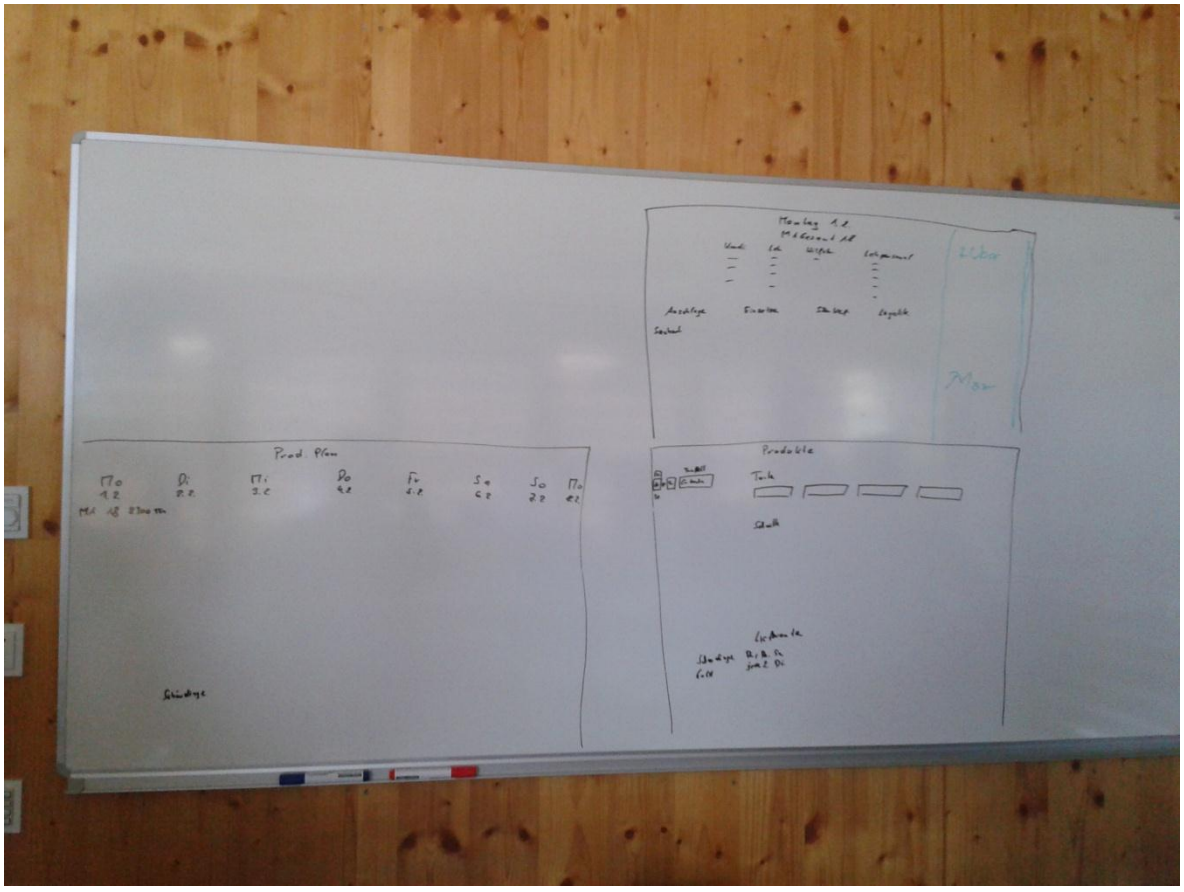

Abb. 5: Erste Skizzierungen beim Kundentreffen

Anhand dieser besprochenen Anforderungen sind ein Lastenheft und erste Design-Entwürfe erstellt worden. Diese wurden dann beim nächsten Treffen, am 2. April 2013, dem Kunden vorgestellt. Hier kam es dann noch zu einigen Abänderungen der Projektanforderungen.

Anschließend wurde ein Pflichtenheft erstellt, welches dann am 16. Mai 2013 gemeinsam mit dem Werkvertrag nach meheren Abänderungen unterschrieben wurde.

Nach dem Unterzeichnen des Pflichtenheftes begann dann die eigentliche Planungs- bzw. Entwicklungsphase. In Abständen von 1-2 Monaten gab es dann weitere Treffen mit unserem Kunden, um zu verhindern, dass das Projekt in eine falsche Richtung verläuft.

Am 17. November 2013 wurde dann die erste Version der Software dem Kunden zum Testen übergeben. Es fand auch gleich eine Unterweisung statt.

Nach einigen kleinen Änderungen bzw. Anpassungen wurde dann am 10. Februar 2014 die Software mit einer Bedienungsanleiung übergeben und somit das Projekt abgeschlossen.

7. Gelerntes

Wenn wir dieses Projekt jetzt noch einmal durchführen würden, würden wir die Projektplanung um vieles besser machen indem wir vor allem genauer und detailierter planen.

Das Problem bei diesem Projekt war, dass uns viele Probleme, die bei einem solchen Projekt auftreten können, noch garnicht bewusst waren. Auch im Pflichtenheft würden wir die Anforderungen viel genauer definieren, damit keine Möglichkeiten offen gelassen werden.

Ein Problem, das dann auftreten kann, ist zum Beispiel, dass der Kunde immer wieder Änderungen vornehmen will, welche im Pflichtenheft nicht genau dokumentiert wurden. Jetzt würden wir von unserem Kunden im Vorfeld schon viel genauere Details erfragen.

7.1. Events

7.1.1. Überall-App Kongress

Der Überall-App Kongress im METAStadt, 1220 Wien hat unserer App und unserem Kunden sehr viel Wissen über die Konkurrenz, Verfahren in der Entwicklung und generell mehr Einblick in die „Materie“ der Appentwicklung und dem Marketing ringsherum.

Wie man auch in der Zeiterfassung sehen kann, wurde danach sofort das Layout verworfen, weil die großen Studios der App-Szene neue Designkonzepte vorgestellt haben und uns diese optisch sehr gefielen. Ebenso war durch die vorgestellten Konzepte eine deutliche Steigerung der Effizienz zu erkennen und ein durchdachterer Ansatz in der Entwicklung.

Das Ticket ist bei einem Gewinnspiel der futurezone.at gewonnen worden und berechtigte einen von uns zu dem Kongress zu fahren. Es wurden wichtige Kontakte geknüpft und Informationen ausgetauscht. Es waren ein paar Vertreter der „Austrian Startups“ vor Ort, die mit uns auch das Treffen im Sektor5 organisierten.

7.1.2. Kod.io Linz

Durch ein Gewinnspiel, das durch die „Austrian Startups“ durchgeführt wurde, sind wir (diesmal alle 2) eingeladen worden, an der „Kod.io Developer Conference“ in Linz teilzunehmen und unser Vorhaben (kurz in englischer Sprache) vor einem mittelgroßen, internationalen Publikum zu präsentieren.

In Startup-Kreise ist das als „pitchen“ bekannt und erklärt ein Vorhaben eines angehenden Unternehmens in möglichst einfacher Sprache und das möglichst kurz.

Der Besuch brachte uns tolle Einblicke in die Welt des Entwickler-Anwerbens und der Unternehmensplanung bzw. in die Organisation eines Events und deren Sprecher/Vortragenden.

Abb. 6: v.l.n.r.: Martin Weise, Martin Thoma vor dem Ars Electronica Center in Linz

8. Ausblick

Da diese Software nicht nur für unseren Kunden, sondern auch für einen Weiterverkauf an andere Kunden gedacht ist, haben wir die Software in einzelne sinnvolle Module zerlegt, aus denen ein anderer Kunde wählen kann. Der Weiterverkauf der Software ist im Pflichtenheft und im Werkvertrag geregelt. Sollten wir mit unserem Projekt weitere Kunden gewinnen können, werden diese dann natürlich andere Anforderungen stellen, so entstehen dann wieder neue Module aus denen dann wieder andere Kunden wählen können.

9. Bedienung

9.1. Login, Menü

Die Startseite ist das Login-Fenster (siehe unten). Hier können Sie sich mit Ihrer Email-Adresse bzw. Ihrem Benutzernamen und Ihrem Passwort einloggen. Falls Sie Ihr Passwort vergessen haben, muss der Administrator ein neues Passwort festlegen unter „Einstellungen“ → „Konten“.

Abb. 7: Planify Login

Sind die Login-Daten korrekt gelangen Sie zum Wochenplan.

Mit dem Button links oben öffnen Sie das Menü. Hier können Sie zu Wochenplan, Dienstplan, Produktion und Einstellungen navigieren und sich ausloggen. Wenn Sie im Wochenplan oder im Dienstplan auf das Datum eines Tages klicken gelangen Sie zum zugehörigen Tagesplan.

9.2. Dienstplan

Zum Dienstplan gelangen Sie über die Menüleiste links. Durch klicken auf das Auge links neben dem Datum eines Tages gelangen Sie zum zugehörigen Tagesplan. Mit den Pfeilen rechts und links neben der Überschrift „Dienstplan“ können Sie eine Woche vor oder zurück springen.

Alle Mitarbeiter sind immer fix eingeteilt und müssen aus der Liste gezogen werden, wenn diese frei haben oder krank sind. Wenn Sie auf den Pfeil rechts neben dem Datum klicken werden alle Mitarbeiter angezeigt die nicht eingeteilt sind. Unter den Mitarbeitern sehen Sie die Mitarbeiter-Minuten die an diesem Tag zur Verfügung stehen.

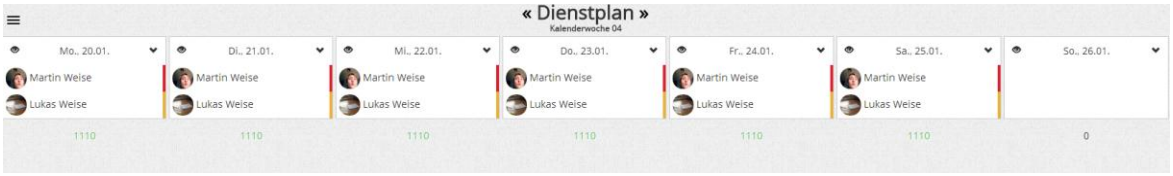

Abb. 8: Planify Dienstplan

9.2.1. Leiharbeiter hinzufügen

Über den „Leiharbeiter“-Button rechts unten im Dienstplan können Sie einen Leiharbeiter hinzufügen. Sie müssen einen Namen eingeben, eine Firma auswählen, welche unter Einstellungen→Mitarbeiter→Firmen vordefiniert werden müssen, eine Start- bzw. Endzeit angeben (die Mittagspause wird dabei nicht berücksichtigt) und die Wochentage an denen der Leiharbeiter in der aktuellen Woche arbeitet.

Mit dem + rechts unten können Sie einen weiteren Leiharbeiter hinzufügen.

Anschließend müssen Sie auf „Einteilen“ klicken um die Leiharbeiter in der aktuellen Woche einzuteilen. Diese werden dann im Dienstplan angezeigt.

Abb. 9: Planify Leiharbeiter

9.3. Wochenplan

Zum Wochenplan gelangen Sie über die Menüleiste links. Durch klicken auf das Datum eines Tages gelangen Sie zum zugehörigen Tagesplan. Über dem Datum sehen Sie die Mitarbeiter-Minuten, die Sie an diesem Tag zur Verfügung haben und darüber die Minuten, welche für die Produktion der unten stehenden Produkte benötigt werden. Mit den Pfeilen rechts und links neben der Überschrift „Wochenplan“ können Sie eine Woche vor oder zurück springen. Mit dem „Heute“- Button rechts unten gelangen Sie zum aktuellen Tag. Durch klicken auf den „Dienstplan“-Button gelangen Sie zum Dienstplan.

Rechts unter Bestellungen sind alle Produkte aufgelistet, welche unter „Produktion“ bestellt worden sind. Durch ziehen auf einen Tag wird dieser als Starttag des Produkts festgelegt und alle Arbeitsschritte teilen sich entsprechend auf die anderen Tage auf. Wenn Sie einen Produktschritt verschieben richtet sich das ganze Produkt danach. Um eine Produktkette zu zerreißen müssen Sie auf den Produktschritt, welchen Sie aus der Kette reißen wollen, klicken und halten bis der Produktschritt sich bewegt. Anschließend müssen Sie den Arbeitsschritt nur zum gewünschten Tag schieben. Wenn Sie ein Produkt löschen wollen müssen Sie es anklicken und anschließend auf den roten „Papierkorb“ rechts unten klicken. Dann erscheint das gelöschte Produkt wieder rechts in der Liste. Wenn Sie es noch einmal anklicken und auf den „Papierkorb“ klicken ist es endgültig

gelöscht. Durch doppelklicken auf den Arbeitsschritt eines Produktes können Sie die Produktmenge direkt im Wochenplan ändern.

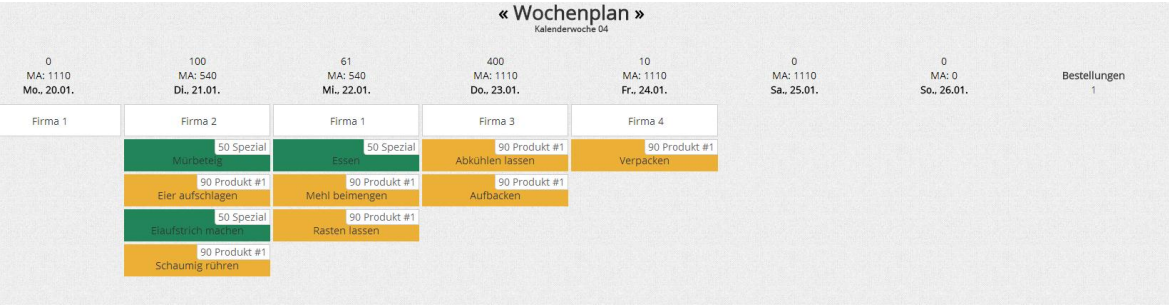

Abb. 10: Planify Wochenplan

9.4. Tagesplan

Zum Tagesplan gelangen Sie indem Sie im Wochenplan auf das Datum des gewünschten Tages klicken oder indem Sie im Dienstplan auf das Auge links neben dem Datum des gewünschten Tages klicken. Durch klicken auf die Pfeile rechts und links neben der Überschrift „Tagesplan“ springen Sie einen Tag vor oder zurück.

Unter jedem Posten sind die Arbeitsschritte aufgelistet, welche an diesem Tag zu erledigen sind. Die Lieferanten, bei denen an diesem Tag bestellt werden muss sind auch aufgelistet. Sie können die Arbeitsschritte auch zu einem anderen Posten schieben. Rechts befindet sich eine Liste mit allen Mitarbeitern, die an diesem Tag Dienst haben. Diese müssen zum entsprechenden Posten gezogen werden um Sie einzuteilen. Unter den Arbeitsschritten wird angezeigt wie viele Bäcker, Hilfsarbeiter, Lehrlinge und Leiharbeiter eingeteilt sind. Darunter sehen Sie die Minuten, welche für die Arbeitsschritte benötigt werden und darunter die verfügbaren Mitarbeiter-Minuten. Wenn Sie aus den eingeteilten Mitarbeitern einen Verantwortlichen für eine Abteilung festlegen wollen müssen Sie auf diesen Mitarbeiter klicken und halten bis neben dem Mitarbeiter ein Herz erscheint.

Wenn Sie die Produktmenge im Tagesplan ändern wollen müssen Sie auf einen Produktschritt klicken und halten bis die Box zum Ändern erscheint.

Um die Arbeitsschritte für alle Posten auszudrucken müssen Sie auf den Button „Drucken“ rechts unten klicken.

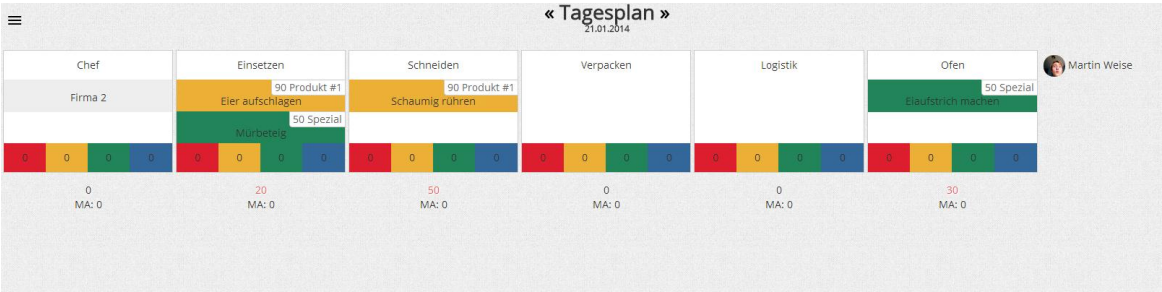

Abb. 11: Planify Tagesplan

9.5. Produktion

Zum Fenster „Produktion“ gelangen Sie über die Menüleiste. Hier sind alle Produkte aufgelistet und nach Produktgruppen sortiert. Sie können hier die unter „Einstellungen“ vordefinierten Mengen auswählen. Anschließend müssen Sie rechts unten auf „Bestellen“ klicken um diese dann im Wochenplan unter Bestellungen anzuzeigen.

Abb. 12: Planify Produktion

9.5.1. Spezialbestellung

Durch Klicken auf den Button „Spezialbestellung“ rechts unten im Produktions-Fenster gelangen Sie in das Spezialbestellungen-Fenster. Hier müssen Sie einen Produktnamen eingeben und entscheiden, ob Sie die Spezialbestellung speichern oder verwerfen wollen und dann noch eine Farbe auswählen.

Wenn Sie darunter auf das + klicken öffnet sich ein neuer Produktschritt. Dieser braucht einen Tag in der Produktkette, einen Namen, einen festgelegten Posten und eine Zeit. Darüber müssen Sie noch eine Menge eingeben. Sie können auch Zutaten eingeben, müssen Sie aber nicht. Mit dem + neben der Menge können Sie eine weitere Menge hinzufügen. Sie müssen für jeden Arbeitsschritt genauso viele Zeiten angeben, wie das Produkt Mengen hat. Durch klicken auf „Neuer Schritt“ können Sie einen weiteren Arbeitsschritt hinzufügen.

Wenn Sie rechts unten auf „Bearbeiten“ klicken können Sie einzelne Mengen oder Produktschritte wieder löschen indem Sie auf den zugehörigen roten „Papierkorb“ klicken. Durch klicken auf Speichern wird die Spezialbestellung im Wochenplan angezeigt.

Wenn Sie eine Spezialbestellung duplizieren wollen müssen Sie die angelegte Spezialbestellung speichern und anschließend unter Einstellungen → Produkte duplizieren.

Abb. 13: Planify Spezialbestellung

9.6. Einstellungen

Zum Fenster „Einstellungen“ gelangen Sie über die Menüleiste. Unter Einstellungen können Sie in 5 Fenstern verschiedene Daten verwalten. Das sind Folgende: Produkte, Mitarbeiter, Lieferanten, Posten, Konten

9.6.1. Produkte

Unter „Produkte“ sind alle vorhandenen Produkte aufgelistet und sortiert nach Produktgruppen. Durch klicken auf ein Produkt öffnet sich ein neues Fenster, in dem Sie das Produkt bearbeiten können. Durch klicken auf das + rechts unten können Sie ein neues Produkt erstellen.

Abb. 14: Planify Einstellungen Produkte

Wenn Sie unter dem neuen Produkt auf das + klicken öffnet sich ein neuer Produktschritt. Dieser braucht einen Tag in der Produktkette, einen Namen, einen festgelegten Posten und eine Zeit. Darüber müssen Sie noch eine Menge eingeben. Sie können auch Zutaten eingeben, müssen aber nicht. Mit dem + neben der Menge können Sie eine weitere Menge hinzufügen. Sie müssen für jeden Arbeitsschritt genauso viele Zeiten angeben, wie das Produkt Mengen hat. Durch klicken auf „Neuer Schritt“ können Sie einen weiteren Arbeitsschritt hinzufügen.

Wenn Sie rechts unten auf „Bearbeiten“ klicken können Sie einzelne Mengen oder Produktschritte wieder löschen indem Sie auf den zugehörigen roten „Papierkorb“ klicken. Durch klicken auf „Speichern“ wird das Produkt gespeichert.

Wenn Sie ein Produkt duplizieren wollen, dann müssen Sie auf das Produkt klicken, welches Sie duplizieren wollen. Anschließend ändern Sie den Produktnamen und klicken auf duplizieren. Nun müssen Sie nur noch die Daten des duplizierten Produkts bearbeiten.

Abb. 15: Planify Einstellungen Neues Produkt Initialansicht

9.6.2. Produktgruppen

Durch klicken auf „Produktgruppen“ rechts unten unter Einstellungen → Produkte gelangen Sie in ein neues Fenster, wo Sie die Produktgruppen bearbeiten, entfernen oder neue hinzufügen können.

Durch klicken auf das + rechts unten können Sie eine neue Produktgruppe hinzufügen. Wenn Sie rechts unten auf „Bearbeiten“ klicken können Sie einzelne Produktgruppen löschen. Um die Produktgruppen zu speichern müssen Sie rechts unten auf „Speichern“ klicken. Der Produktgruppe, welche mit dem ausgefüllten Stern markiert ist, werden die Spezialbestellungen zugewiesen, aber nur denen die auch gespeichert werden sollen.

Abb. 16: Planify Einstellungen Produktgruppen

9.6.3. Mitarbeiter

Unter „Mitarbeiter“ sind alle vorhandenen Mitarbeiter aufgelistet. Hier können Sie die vorhandenen Mitarbeiter bearbeiten. Wenn Sie auf das + rechts unten klicken wird ein neuer Mitarbeiter hinzugefügt.Jetzt muss noch Vorname, Nachname, Geburtsdatum, Berufsgruppe, Arbeitszeit(Beginn & Ende!! Mittagspause nicht berücksichtigt!!)

eingegeben werden. Sie können auch ein Bild des Mitarbeiters machen & dieses mit einem Klick auf das Bild hochladen. Besitzen Sie ein Android oder iOS-Tablet, können Sie auch direkt ein Bild mit der Kamera machen, das automatisch hochgeladen wird.

Anschließend müssen Sie rechts unten auf „Speichern“ klicken, damit die Änderungen gespeichert werden. Wenn Sie einen Mitarbeiter löschen wollen müssen Sie rechts unten auf „Bearbeiten“ klicken und anschließend auf den „Papierkorb“ unter dem zu löschenden Mitarbeiter.

Abb. 17: Planify Einstellungen Mitarbeiter

Wenn Sie rechts unten auf den Button „Firmen“ klicken können Sie die Leihfirmen bearbeiten. Hier sind alle Leiharbeiterfirmen aufgelistet und können bearbeitet werden. Wenn Sie auf das + rechts unten klicken können Sie eine neue Firma hinzufügen. Wenn Sie eine Firma löschen wollen müssen Sie auf „Bearbeiten“ klicken und anschließend auf „Löschen“ rechts neben der Firma. Dann müssen Sie rechts unten auf „Speichern“ klicken, damit die Änderungen gespeichert werden.

Abb. 18: Planify Einstellungen Leiharbeiterfirmen

9.6.4. Lieferanten

Unter „Lieferanten“ sind alle vorhandenen Lieferanten aufgelistet. Hier können Sie die vorhandenen Lieferanten bearbeiten. Wenn Sie auf das + rechts unten klicken wird ein neuer Lieferant hinzugefügt. Jetzt muss noch der Firmenname, ein Bestellungsintervall, ein Startdatum und die Wochentage, an denen bestellt werden muss, eingegeben werden. Anschließend müssen Sie rechts unten auf „Speichern“ klicken, damit die Änderungen gespeichert werden. Wenn Sie einen Lieferanten löschen wollen müssen Sie rechts unten auf „Bearbeiten“ klicken und anschließend auf den „Papierkorb“ neben dem zu löschenden Lieferanten klicken.

Einstellungen
Produkte Mitarbeiter Lieferanten Posten Konten
Firma 1 Wöchentlich 14.12.2013 Montag Dienstag Mittwoch Donnerstag Freitag Samstag Sonntag
Firma 2 Wöchentlich 14.12.2013 Montag Dienstag Mittwoch Donnerstag Freitag Samstag Sonntag
Firma 3 Wöchentlich 14.12.2013 Montag Dienstag Mittwoch Donnerstag Freitag Samstag Sonntag
Firma 4 Wöchentlich 15.12.2013 Montag Dienstag Mittwoch Donnerstag Freitag Samstag Sonntag

Abb. 19: Planify Einstellungen Lieferanten

9.6.5. Posten

Unter „Posten“ sind alle vorhandenen Posten aufgelistet. Hier können Sie die vorhandenen Posten bearbeiten. Wenn Sie auf das + rechts unten klicken wird ein neuer Posten hinzugefügt. Anschließend müssen Sie rechts unten auf „Speichern“ klicken, damit die Änderungen gespeichert werden. Wenn Sie einen Posten löschen wollen müssen Sie rechts unten auf „Bearbeiten“ klicken und anschließend auf den „Papierkorb“ unter dem zu löschenden Posten klicken. Mit dem ausgefüllten Stern in der rechten oberen Ecke markieren Sie den Posten dem alle Lieferanten-Bestellungen im Tagesplan zugewiesen werden.

Abb. 20: Planify Einstellungen Posten

9.7. Konten

Unter „Konten“ sind alle vorhandenen Konten aufgelistet. Hier können Sie die vorhandenen Konten bearbeiten. Wenn Sie auf das + rechts unten klicken wird ein neues Benutzer-Konto hinzugefügt. Sie können dann sehr übersichtlich festlegen, was dieser Benutzer alles sehen darf. Der Benutzer in dieser Abbildung darf die Module „Wochenplan“ und „Dienstplan“ sehen.

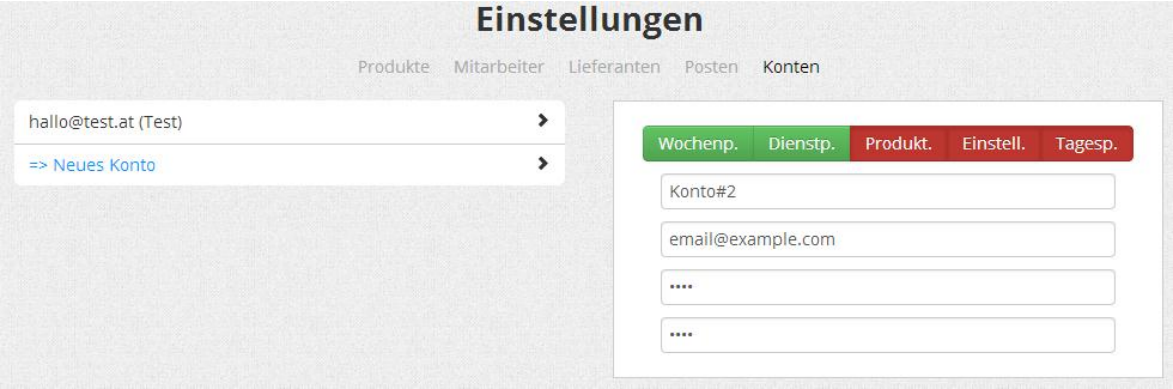

Abb. 21: Planify Einstellungen Konten

Wenn Sie auf ein Konto klicken können Sie die Informationen dieses Kontos eingeben bzw. bearbeiten. Anschließend müssen Sie rechts unten auf „Speichern“ klicken, damit die Änderungen gespeichert werden.

Sie können für jeden Benutzer festlegen, welche Fenster er sehen bzw. bearbeiten darf. Grün berechtigt den Benutzer das Fenster zu bearbeiten, Rot nicht.

10. Programmieren (Theorie)

10.1. Allgemeines[1]

10.1.1. Prinzip

Das World Wide Web basiert auf der klassischen Client-Server-Architektur. Darunter versteht man eine gemeinsame Informationsverarbeitung, welche einzelne Komponenten zwischen Programmen auf miteinander vernetzten Rechnern aufteilt. In diesem Netz können sowohl Server, die über das Netz Dienstleistungen anbieten, als auch Clients teilnehmen, die bei Bedarf solche Dienste anfordern. Die Kommunikation zwischen Server und Client erfolgt in Form von Transaktionen, die ein Verbund aus logisch zusammenhängenden Aktionen ist.

Client und Server sind entweder über ein lokales Netz oder über große Entfernungen hinweg, zum Beispiel über eine Satellitenverbindung miteinander verbunden. Relevante Eigenschaften von Teilnehmern dieses Netzes ist beispielsweise das Rechenleistungsvermögen. Dabei kann der Client den Server leistungstechnisch gesehen durchaus übersteigen, denn die Idee hinter dieser Architektur ist eine optimale Ausnutzung der Ressourcen der beteiligten Systeme für den jeweiligen Zweck.

10.2. Sprachen

10.2.1. HTML

Die Hypertext Markup Language (HTML) wird vom Browser des Client interpretiert und ist eine Auszeichnungssprache, die verwendet wird, um Dokumente im World Wide Web zu strukturieren. Dabei wird immer in Klartext geschrieben und auf Kontrollstrukturen, wie es Bedingungen, Schleifen oder Variablen sind, verzichtet. HTML-Dokumente sind Plattformunabhängig und benötigen einen Browser, um aus dem Quellcode ein nutzbares Web-Dokument zu generieren. HTML wird von dem W3C standardisiert und stetig verbessert. Die Aktuelle Version trägt die Nummer 4.01 und wird in Entwicklerkreisen nur noch eingeschränkt verwendet.

[Sei06]

Man verwendet den inoffiziellen Standard HTML5, der auf experimenteller Basis viele Funktionen unterstützt, wofür die man sonst andere Sprachen, wie beispielsweise JavaScript benötigen würde.

10.2.1.1. Das W3C

Das Ziel des World Wide Web Consortium ist es Protokolle und Richtlinien zu entwickeln, die ein langfristiges Wachstum des Internets sicherstellen. Dabei ist es den

[1] http://gd.tuwien.ac.at/study/hrh-glossar/1-2_17.htm (03.02.2014)

Mitgliedernwichtig, dass Menschen unabhängig ihrer verfügbaren Hardware, Software, Netzwerk- Infrastruktur, Sprache, Kultur, Heimat, physischen oder mentalen Fähigkeiten das Internet für Kommunikation, Kommerz und der Möglichkeit, Wissen auszutauschen benützen können.[2]

Jedes Mitglied muss sich an folgende Grundsätze halten:

- Entscheidungen werden unter Einhaltung von Gerechtigkeit und Fairness unter den Teilnehmern gefällt. Keine Partei wird bevorzugt oder führt die Entwicklung der Standards an.
- Bei allen Prozessen werden alle Ansichten berücksichtigt und können angesprochen werden, sodass eine Übereinstimmung über verschiedene Interessen gefunden werden kann.
- Wenn ein Standard vorgeschlagen wird, muss dieser öffentlich bekannt gemacht werden, ebenso wie die Entwicklungsaktivitäten, Umfang der durchzuführenden Arbeiten und die Bedingungen für die Teilnahme an den Arbeiten. Materialien, wie zum Beispiel Aufzeichnungen von Entscheidungen müssen leicht zugänglich sein und bereitgestellt werden. Es werden Zeiten vorgesehen, in denen die Öffentlichkeit über den Vorschlag diskutieren kann.
- Die Normungsaufgaben werden nicht ausschließlich von einer Person, Gesellschaft oder Interessengruppe dominiert.
- Die Standardisierungsprozesse sind offen für alle interessierten Parteien.

Die beschlossenen Normen des W3C werden implementiert für alle zugänglich gemacht. Alle Standards des W3C sind freiwillig umzusetzen, der Erfolg wird durch den Markt bestimmt.[3]

10.2.1.2. Syntax

Die Syntax von HTML kann man (sehr vereinfacht) als Strukturierung von Daten innerhalb eines Dokuments bezeichnen. Sie macht im Grunde genommen nichts anderes, als Daten in bestimmte Muster zu packen und diese dann dem Browser zu übergeben. Diese Muster werden Tags genannt und haben üblicherweise die Form `<html></html>`. Wie man erkennt gibt es einen Eröffnungs-Tag (Starting-Tag) und einen Schließenden-Tag (Closing-Tag). Innerhalb dieser Tags können dann mehrere Elemente Platz finden.[4]

```
<[HTML-Element] [Attribut]="[Wert]"></[HTML-Element]>
```

Code 1: HTML-Syntax

Ein nach den Standards des W3C definierten Vorgaben besteht ein konformes HTML-Dokument besteht für gewöhnlich aus vier elementaren Teilen:

[2] http://www.w3.org/Consortium/mission (04.02.2014)
[3] http://open-stand.org/principles/ (04.02.2014)
[4] http://gd.tuwien.ac.at/languages/html/html-design/makepage/ (03.02.2014)

- Dokumenttypdeklaration: Angabe welche HTML-Version verwendet wird. zB: `<!DOCTYPE html>` für HTML5
- Wurzelelement: Das HTML-Wurzelelement `<html></html>`
- Kopfelement: In diesem sind Kopfdaten, wie zB: Titel, CSS Datei-Links, etc. vorhanden: `<head></head>`
- Body: Der anzuzeigende Inhalt eines Dokuments mit Text, Bilder, Videos, etc.

Ein exemplarisches Beispiel:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Mein erstes HTML-Dokument</title>
    </head>
    <body>
        <p>Hallo Welt!</p>
    </body>
</html>
```

Code 2: W3C konformes HTML Grundgerüst

Man gestaltet die Web-Applikation mithilfe sogenannter CSS-Dateien, die man dezentral in das HTML-Dokument einbindet.

10.2.1.3. HTML5[5]

Die nicht-offizielle Version HTML5 bietet Entwicklern eine riesige Auswahl von neuen Technologien, die sich sehr leicht in bestehende Projekte implementieren lassen. Ergänzte Funktionen von HTML5 sind unter anderem:

- Offlinespeicher
- Echtzeit Kommunikation
- Hardwarezugang
- Markupneuerungen
- Bessere Multimedia Unterstützung

Für die Diplomarbeit sind die Punkte „Hardwarezugang“ und „Neues Markup“ relevant, weil es nötig war, die Kamera des Endgerätes anzusteuern. Mit HTML5 ist sichergestellt, dass die Kamera eines jeden Gerätes mit modernen Browser angesteuert werden kann ohne für jedes einzelne Gerät eine spezielle Notation zu tätigen.

Das Markup in dem in Abb. 22 zu sehenden Dokument ist nicht wirklich aufwändig, wie sich vielleicht vermuten lässt. Man kann die Kamera eines Endgerätes sehr leicht durch Erweiterung des `input`-Elements mittels `accept="image/*;capture=camera"` aufrufen, HTML5 fügt das aufgenommene Bild als Datei dem `input`-Feld an.

Für die vorhandenen Screenshots wurde ein iPhone® 4G verwendet. Man kann aber genauso gut mit jedem anderen (mobilen) Endgerät mit integrierter Kamera auf

[5]http://slides.html5rocks.com/ (03.02.2014)

dieeingebaute Kamera zugreifen. Mit dem heutigen Fortschritt (Stand: 16. Februar 2014) ist es jedoch noch nicht einwandfrei möglich, Herstellerübergreifend Fotos mit integrierten/externen Kameras von Laptops oder PCs zu machen. Es ist aber nur eine Frage der Zeit, bis auch das gelöst wird.

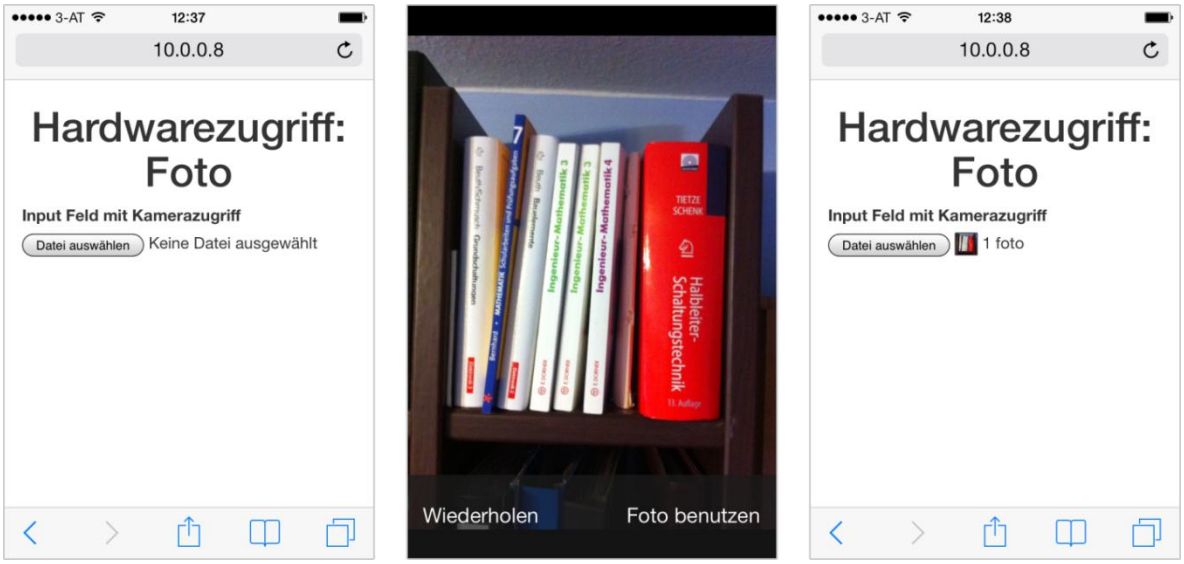

Abb. 22: Zugriff auf Hardware mit HTML5

1.1.2.3. Formulare mit HTML5

Formulare sind wichtig für die Interaktion von Inhalt und dem Nutzer. Sie werden zur Kontaktaufnahme, verifizieren von Login-Daten, angeben persönlicher Daten und Kreditkarteninformationen und vielem mehr verwendet und sind ein wichtiger Bestandteil einer Applikation. Beinahe jede Internetpräsenz benötigt ein Formular zur Interaktion, auch wenn es nur ein Newsletter-Abo ist, wie in Abb. 23 zu sehen.

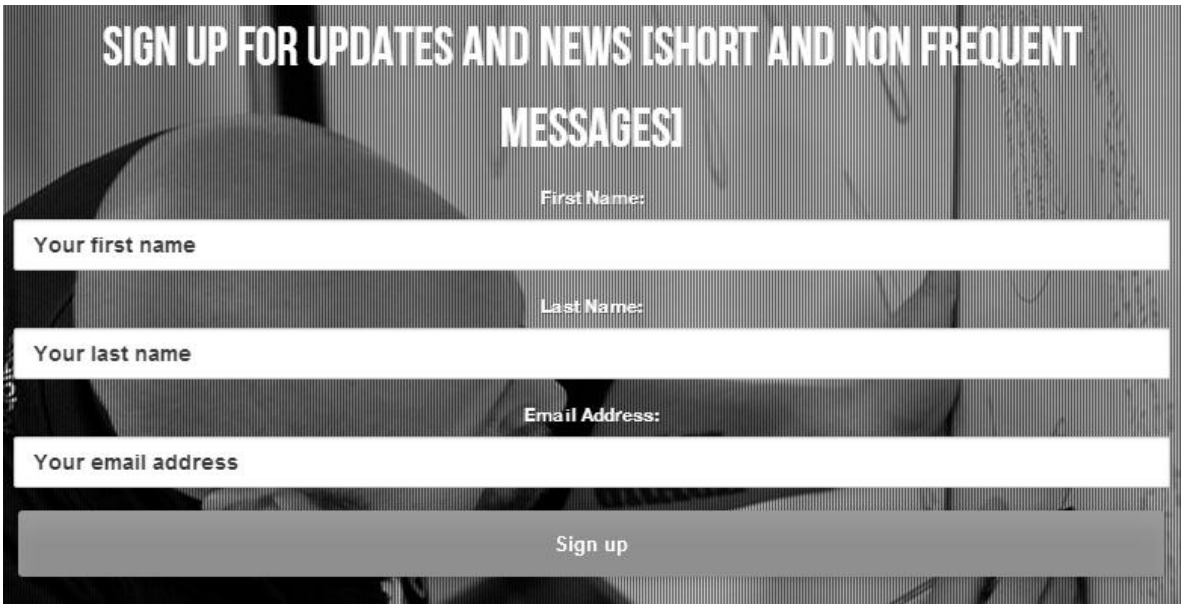

Abb. 23: Formular in HTML5 mit placeholder-Attributen

Trotz der häufigen Verwendung ist die Ausstattung an W3C konformen HTML-Attributen zu dem Thema recht spärlich gewesen. Das erforderte den Einsatz von serverseitigen Skriptsprachen und war nur durch große Umwege möglich, was den Datenverkehr zwischen jeder Eingabe erheblich ansteigen ließ. HTML5 setzt genau dort an, wie der Nutzer das Formular benützen möchte.

Das Platzhalterattribut `placeholder` ist ein gutes Beispiel, um diese Neuerung zu demonstrieren. Wenn man nämlich das `input`-Feld um diese Eigenschaft erweitert und einen Wert anfügt, erscheint im `input`-Feld ein Wert, der beim Schreiben ausgeblendet wird.

Weitere Attribute wären `required`, welche das Feld zu einem verpflichtenden Feld machen und `autofocus`, welcher beim Aufruf eines Dokuments den Cursor direkt in das Feld setzt, man kann sofort in dieses Feld Eingaben tätigen.

Um das in Abb. 23 angegebene E-Mail Feld auch bei mobilen Geräten ohne großen Aufwand eingeben zu können, hat man als Entwickler nun die Möglichkeit dieses Feld zu einem `<input type="email" />` zu machen. Moderne Browser prüfen dieses Feld nun auf eine korrekte E-Mail Syntax. Das heißt – mit anderen Worten – man muss sich nicht um JavaScript Überprüfungen kümmern, der Browser überprüft das Feld eigenständig und sagt dem Nutzer bei falschen Eingaben auch, was falsch ist.

Zudem wird bei mobilen Geräten zwischen verschiedenen Eingabetypen die Tastatur gewechselt, um die Eingabe von Uhrzeiten, Daten, Zahlen, Telefonnummern und dergleichen zu erleichtern bzw. Falscheingaben gar nicht erst möglich zu machen.

Das kann man in Abb. 24 sehr gut sehen. Die Firma Apple® Inc. geht mit hervorragend gutem Beispiel voran und war einer der Ersten Anbieter, der diese Neuerungen mit

Spezialtastaturen unterstützt hat. Sogar ein alter iPod® Touch 2G (Erscheinung 2008) konnte ohne Updates (das letzte erfolgte 2011) moderne HTML5 Seiten anzeigen und wechselte die Tastatur bei den zuvor erwähnten Eingabetypen.

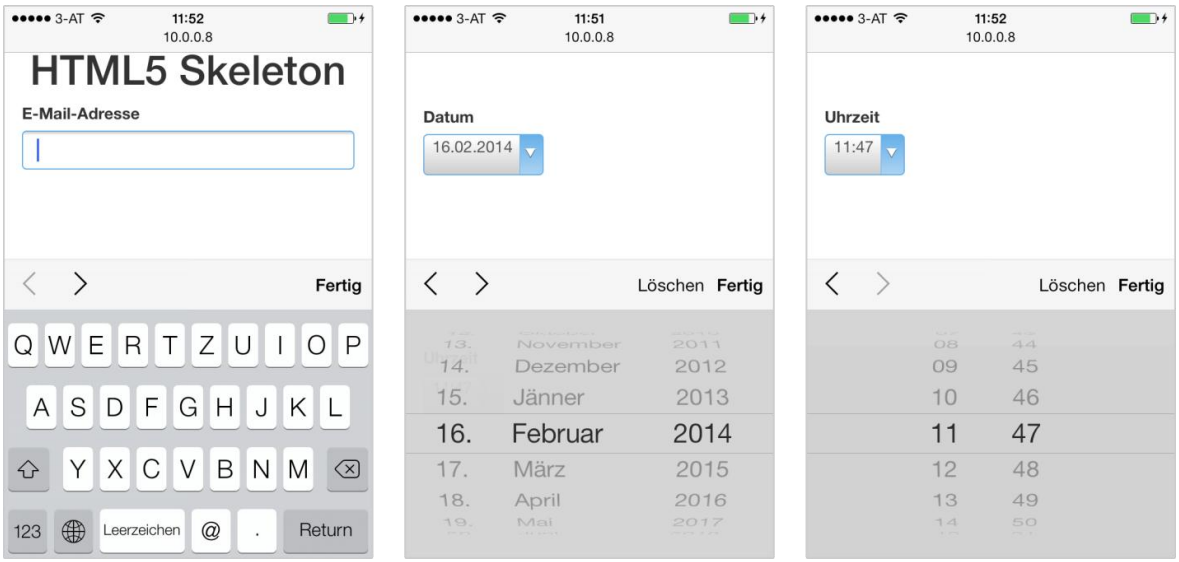

Abb. 24: Tastaturen bei verschiedenen Eingabetypen

[Zil13]

10.2.2. CSS

Ist das vom Server geholte Dokument ein reines HTML-Dokument, welches optisch nicht besonders ansprechend ist, hat man die Möglichkeit die Seite noch mehr seinen Vorstellungen anzupassen. Die Auszeichnungssprache HTML lässt nämlich bis auf kleine „Design"-Vorschläge nicht viel Kreativität zu. Aus diesem Grund wurde die HTML-Ergänzungssprache CSS (Cascading Stylesheets) geschaffen, mit deren Hilfe man die Stilinformationen bestimmter Elemente anhand ihrer Eigenschaften oder ihre Position im Dokument ändern kann.[6]

10.2.2.1. CSS Version 3[7]

Zudem trägt CSS einen maßgeblichen Anteil zur Anpassung der Stilinformationen aller Elemente eines HTML-Dokuments an die Bildschirmbreite oder -höhe bei und ist damit hauptverantwortlich für die dynamische Anzeige von Inhalten auf Geräten verschiedener Größe und Bildschirmauflösung. Die aktuelle Version ist CSS3 und steckt offiziell seit 2000 in der Entwicklung fest, was bedeutet, dass eine Standardisierung durch das W3C noch nicht stattgefunden hat. Dennoch unterstützen viele moderne Browser bereits viele der für die Fertigstellung von CSS3 geplanten Module. Großteils unterstützte Module sind zum Beispiel:

[6] http://de.selfhtml.org/css/intro.htm (14.01.2014)
[7] http://www.w3.org/Style/CSS/current-work (14.01.2014)

- CSS Color Level 3: Bisher konnte man Farben in CSS durch sogenannte `color` keywords (zB: `black`) oder durch ihren hexadezimalen Wert (zB: `#000000`) definieren. Hier ergibt sich eine Einschränkung, welche durch CSS Color Level 3 großteilig aufgehoben wird. Man erweitert um das RedGreenBlue (RGB)- und das HueSaturationLightness (HSL)-Modell.
- Selectors Level 3: Erweiterung der Selektoren und Pseudoklassen aus CSS2.
- Media Queries: Hauptverantwortlich für den Umstand, dass ein und dasselbe HTML-Dokument auf verschiedenen Geräten optimal (anhand ihrer Bildschirm-dimension, -proportion, etc.) dargestellt wird.

Es wurde CSS Version3 verwendet, da sie viele Neuerungen mit sich bringt und die Web-Applikation wesentlich ansehlicher macht. Außerdem ist die Unterstützung durch die Browser schon sehr fortgeschritten, sodass man viele Funktionen bereits ohne große Konfigurierung nützen kann.

10.2.2.2. Einsatz

Um das aus Code 1 bekannte Paragraph-Element zu stilisieren kann man zum Beispiel den Wert des `style`-Attributs ändern:

```
<p style="color:red;">Ich bin ein Absatz mit roter Schrift.</p>
```

Code 3: HTML-Paragraph mit inline CSS

Man kann sich vorstellen, dass beim Ändern der einzelnen Stilinformationen bei mehreren Elementen der Wartungsaufwand den Nutzen übersteigt. Die Stilinformationen für jedes einzelne Element im HTML-Quellcode vorzunehmen ist auch nicht die Intention der CSS Entwickler Håkon Wium Lie[8] und Bert Bos.

10.2.2.3. Nutzen

Die beiden wollten mit ihrer neu entwickelten Stylesheet-Sprache Stilregeln aufstellen, die an Elemente vererbt werden konnten. Sie legten damit den Grundstein für die Auslagerung aller Stilinformationen von einem HTML-Dokument in ein Cascading Stylesheet, einer CSS-Datei. Dort lassen sich Stilinformationen zentral sammeln. Sie minimieren damit den Wartungs-aufwand enorm und tragen zu einer fortschrittlichen Gestaltung des Internets bei. Wie in Code 5 zu sehen ist, gestaltet sich die Syntax in CSS sehr einfach und mit wenig Schreibarbeit.

10.2.2.4. Verwendung

Um CSS benützen zu können werden keinerlei Installationen benötigt. Ein einfacher Texteditor, der bereits mit einer Windows-Installation am PC vorhanden ist, wie editor.exe (*engl.:* notepad.exe), reicht völlig aus.

[8] http://people.opera.com/howcome/2006/phd/ (29.04.2014)

Die ausgelagerte CSS-Datei kann nun in dem HTML-Dokument verbunden werden. Dafür gibt es einen eigenen Tag, den man einfach irgendwo in dem <head></head>-Bereich einträgt.

```
<link rel="stylesheet" type="text/css" href="file.css" />
```

Code 4: Einbinden der CSS-Datei „file.css“ in ein HTML-Dokument

Dieser Tag ist anders als die vorher erwähnten. Diese Art von Tags nennt man Self-Closing, also selbstschließend. Es gibt somit keinen Closing-Tag, der das Element schließt, aufgrund dieser Tatsache gibt es auch keinen Inhalt. Das Element wird nämlich rein durch seine Attribute rel, type und href bestimmt. Es soll auch keinen Inhalt geben.

10.2.2.5. Syntax[9]

```
p {
    color: red;
}
```

Code 5: CSS Code

In CSS hat diese Form einen bestimmten Namen und auch eine Form, die immer dieselbe ist. Auf Code 5 bezogen ergibt sich ein p-Selektor, der die color-Eigenschaft aller p-Elemente auf red setzt. Neben diesem Element-Selektor gibt es noch vier weitere Methoden den Selektor zu beschreiben:

- Eindeutige Kennzeichnung: Man kann durch Auswahl eines eindeutigen Elements, wie zum Beispiel durch das HTML-Markup-Attribut id, dessen Wert im Dokument einzigartig sein muss. Diese Bezeichnung sollte die Bedeutung des Elements im Dokument bezeichnen, wie beispielsweise `#passwort { color: blue; }`
- Klassen: Über das HTML-Markup-Attribut class lassen sich für gewöhnlich sehr effizient mehrere Elementgruppen eines Dokuments anwählen, wie durch das folgende Beispiel: `.form-control{}`. Das ist auch Sinn & Zweck dieses Selektortyps.
- Universal: Durch den Einsatz eines Wildcard-Symbols, wie beispielsweise * kann man allen Elementen eines Dokuments bestimmte Eigenschaften zuweisen.
- Attribut: Ebenso ist es möglich durch HTML-Attribute, wie href, src, type, etc. bestimmte Elemente oder Elementgruppen auszuwählen. Das ist durch den reinen Attributnamen (zB: `[type]`) oder auch durch einen Zuweisungsoperator (zB: `[type="text"]`) möglich.
- Pseudo-Klassen bzw. -Elemente: Es ist möglich bestimmte Elemente mittels ihrer Position oder anderen Eigenschaften zu selektieren, ohne dabei den Quelltext ändern zu müssen. Um nur das Erste p-Element auszuwählen, schreibt man: `p:first-child {}`

[9] http://de.wikibooks.org/wiki/Websiteentwicklung:_CSS:_Syntax (28.01.14)

10.2.2.6. Bootstrap[10]

Für diese Arbeit, wird ein von der Firma Twitter Inc. entwickeltes CSS-Framework, also ein CSS-Gerüst verwendet, welches über definierte Elementklassen (`class`) aktiviert wird. Man kann Bootstrap kostenlos herunterladen[11] und den Ordner `dist` (Version 3) in das Wurzelverzeichnis des Projektes verschieben.

Abb. 25: Bootstrap Logo

Diese einfache, als auch effektive Handhabung ist mitverantwortlich für den bahnbrechenden Erfolg des ursprünglich firmenintern gestarteten Projekts. Auf die Diplomareit bezogen war es möglich den Fokus der Programmierung weniger auf das Design der Elemente zu setzen, sondern auf die Funktionalität der Anwendung. Das heißt man muss sich nur sehr wenig um das Design einer Web-Applikation kümmern, das ist ein enormer Vorteil gegenüber Projekten ohne CSS-Framework. Die Farben und Größen der Elemente harmonieren automatisch sehr gut miteinander.

10.2.3. JavaScript

In allen modernen Web-Applikationen lassen sich heute Teile von JavaScript finden. Es ist eine Skriptsprache, die hauptsächlich beim Client interpretiert und ausgeführt wird. Es ist bereits in jedem modernen Webbrowser integriert und muss nicht erst gekauft oder installiert werden.

10.2.3.1. DOM[12]

Um JavaScript benützen zu können, muss man das Document Object Model (DOM) verstanden haben. Beim Laden eines HTML-Dokuments vom Server erstellt der Browser des Client das DOM, das eine Gitterartige Beschreibung des Aufbaus des Dokuments ist.

[10] http://getbootstrap.com/about/ (14.01.2014)
[11] http://getbootstrap.com (29.04.2014)
[12] http://www.w3schools.com/js/js_htmldom.asp (29.01.2014)

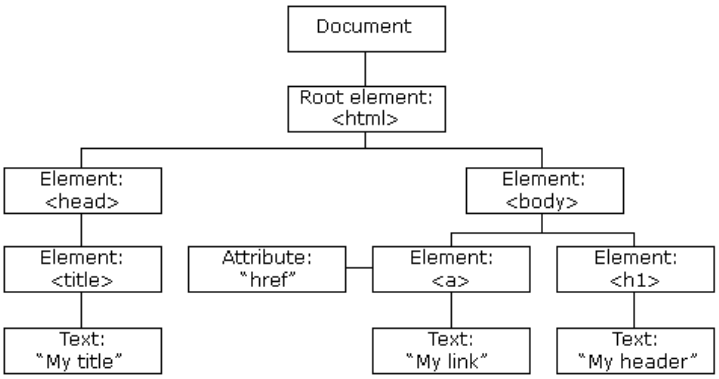

Abb. 26: Das Document Object Model (DOM)

Das DOM ist plattform- und sprachenunabhängig und bildet einen gemeinsamen Zugangspunkt, um den Inhalt, Struktur und Stil eines Dokuments dynamisch zu lesen und zu ändern. Wie in Abb. 26 zu sehen, ist die Wurzel eines Dokuments das `<html>`-Tag, der sich in das Kopfelement `<head>` und Körperelement `<body>` aufspaltet. Diese Elemente unterteilen sich dann inhaltsabhängig in weitere Elemente.

10.2.3.2. Syntax

Die Syntax erinnert stark an die von C oder die des Namensvetters Java. Die beiden haben jedoch nur mäßige Gemeinsamkeiten, Java muss vor dem Verwenden nämlich von dem Prozessor kompiliert werden und zählt somit zu den Programmiersprachen. JavaScript hat im Gegensatz zu seinem Namensvetter weitaus mehr Einschränkungen und kann zum Beispiel auch nicht auf Systembezogene Abteile zugreifen, es ist also nicht möglich Systembezogene Dateien mit JavaScript zu lesen oder zu beschreiben, was die Systemsicherheit einigermaßen gewährleistet.

Man kann JavaScript zwar direkt in ein HTML-Dokument einbinden, jedoch ist das nicht Zweckmäßig, es wird daher ein seperates JavaScript Dokument verfasst, welches `common.js` genannt wird und über `<script src="common.js"></script>` in das HTML-Dokument eingebunden wird.

Möchte man ein Skript realisieren, das den Besucher fragt, wie er heißt und ihm ein `input`-Feld, sowie einen `button`, der beim Klicken den Namen des Besuchers ausgibt, muss man das Markup aus Code 6 und das passende Skript aus Code 7 verwenden.

```
<h1 class="text-center">Hallo!</h1>
<form onsubmit="javascript:check();">
    <label for="name">Wie ist dein Name?</label>
    <div class="input-group">
        <input type="text" class="form-control" id="name" />
        <span class="input-group-btn">
            <input type="submit" class="btn btn-primary" value="Weiter">
        </span>
    </div>
</form>
```

Code 6: HTML Code zum Programm in Abb. 27

```
function check() {
	var obj = document.getElementById("name");
	if(obj.value.length > 0)
		alert("Hallo "+ obj.value +"!");
}
```

Code 7: JavaScript zum Programm in Abb. 27

Die Funktion `check()` fragt zuerst das Element mit der `id="name"` an. Danach wird der Inhalt dieses Elements auf seine Länge geprüft, ist diese größer als 0, also wurde etwas eingegeben, dann wird der Besucher mit einer sog. `alert`-Box benachrichtigt.

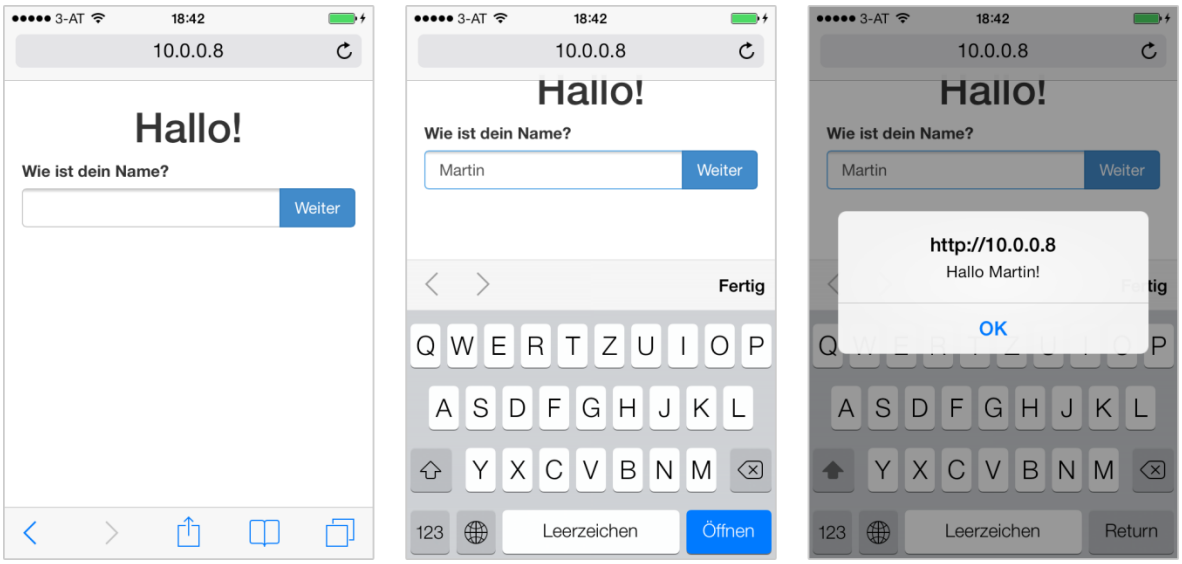

Abb. 27: Beispielanwendung unter Verwendung von JavaScript

10.2.3.3. jQuery

Für JavaScript gibt es, wie bei vielen anderen hier aufgeführten Sprachen, Bibliotheken, welche die Entwicklung erheblich beschleunigen und den Code wesentlich weniger wartungsaufwändig machen. Für die Diplomarbeit wurde die populärste JavaScript Bibliothek, jQuery, benützt, da die Dokumentation am ausführlichsten ist und sie mit jQuery Mobile erweitert werden kann, was den Einsatz der Web-Applikation auf mobilen Geräten noch besser macht.

Um den Unterschied von JavaScript und jQuery zu verdeutlichen, sei folgende Aufgabenstellung gegeben: Schreibe ein HTML-Dokument und ein passendes Skript, welches die Arbeitszeit anhand von Startzeit und Endzeit berechnet (ohne Mittagspause).

```
<h1 class="text-center">Dienstzeit</h1>
<div>
    <label for="start">Antritt</label>
    <input type="time" class="form-control" id="start" />
</div>
<div>
    <label for="end">Feierabend</label>
    <input type="time" class="form-control" id="end" />
</div>
<div>
    <button class="btn btn-lg btn-primary"
id="sbmtBtn">Berechnen</button>
</div>
```

Code 8: Markup zur Aufgabenstellung

Das Markup zur Aufgabenstellung bleibt für beide Skripts gleich und lässt sich in Code 8 betrachten. Es gibt zwei `input`-Felder vom Typ `time` und einen `button`, der dann das JavaScript über einen sog. Event-Handler, also eine definierte Funktion beim Interagieren mit dem Button aufrufen soll. Dazu wird lediglich die `id` des `button` benötigt.

```
var obj = document.getElementById("sbmtBtn");
obj.addEventListener("click", calc, false);
function calc() {
    var start = document.getElementById("start");
    var end = document.getElementById("end");
    start = new Date("01/01/2014 " + start.value).getHours();
    end = new Date("01/01/2014 " + end.value).getHours();
    alert("Du wirst "+(end-start)+" Stunden arbeiten! (Ohne
Mittagspause)");
}
```

Code 9: Lösung in JavaScript

Wie in Code 9 zu sehen ist, wird dem Objekt `obj` ein sog. `EventListener` zugewiesen, der beim Klicken auf `obj` die Funktion `calc` aufruft. Diese berechnet die Start- & Endzeit des Mitarbeiters anhand der Differenz der absoluten Stunden eines beliebigen Datums. Hierbei sei noch gesagt, dass der Verweis auf `common.js` erst am Ende des HTML-Dokuments stehen darf, steht er im `head`-Bereich des HTML-Dokuments, wird das Skript vor dem Interpretieren des Inhalts, also des DOMs aufgerufen, und somit ist ein Interagieren (zB. dem Hinzufügen eines EventListeners) unmöglich!

```
$(document).ready(function(){
    $('#sbmtBtn').on('click',function(){
var start = $('#start').val();
var end = $('#end').val();
        start =new Date("01/01/2014 "+ start).getHours();
        end =new Date("01/01/2014 "+ end).getHours();
alert("Du wirst "+(end-start)+" Stunden arbeiten (Ohne
Mittagspause!)");
});
});
```

Code 10: Lösung mit jQuery Bibliothek

Man sieht sofort, dass in Code 10 auf ein Bereitsein des Dokuments gewartet wird. Das heißt, man kann das Skript auf im `head`-Bereich des HTML-Dokuments einbinden, es wird keinen Fehler durch das DOM geben. Gleich danach wird dem Element mit der `id` (ein `#`-Zeichen in jQuery) `sbmtBtn` der Klick-Eventhandler zugewiesen. Der Rest ist gleich mit Code 9. Das Ergebnis wird in Abb. 28 dargestellt.

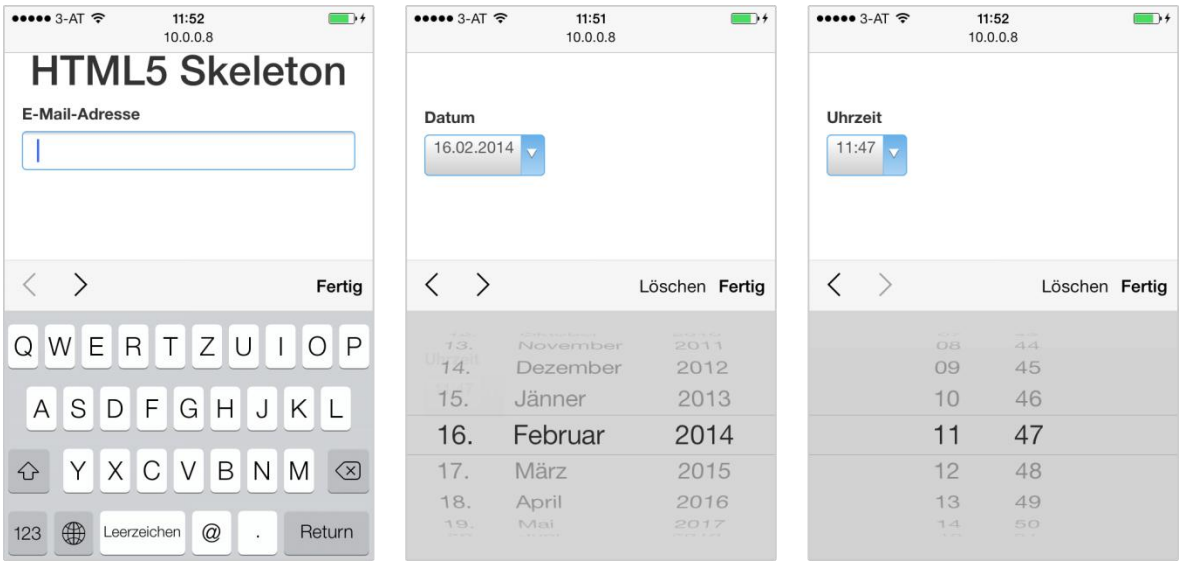

Abb. 28: Ablauf der Aufgabenstellung

In Abb. 28 fragt der Benutzer seine Arbeitsstunden ab. Als Antrittszeit wird 8:00 angegeben und der Feierabend beginnt um 17:15. Das Programm berechnet (völlig korrekt) eine Arbeitszeit von 9 Stunden ohne Pause. Selbst, wenn man bis 17:59 arbeitet, müsste man eigentlich 10 Stunden (ohne Pause) arbeiten, das Skript sagt jedoch 9 Stunden. Das rührt aus der `.getHours()`-Funktion. Diese nimmt keine Rücksicht auf die eingegebenen Minuten.

10.2.3.4. Einsatz

JavaScript kann ohne Installation mit jedem modernen Browser ausgeführt werden. Möchte man hingegen eine Bibliothek verwenden, wie jQuery eine ist, muss man diese vorher herunterladen[13] und im HTML-Dokument einbinden. Es ist empfehlenswerter „production"-Versionen herunterzuladen, weil diese Ladezeit sparen und damit etwas effizienter, als die unkomprimierte „development"-Versionen, arbeiteten.

Die heruntergeladene `jquery-x.x.x.min.js` kann nun im `dist/js/` Ordner von Bootstrap untergebracht werden und im HTML-Dokument eingebunden werden.

[13] http://jquery.com/download/ (29.04.14)

10.2.3.5. AJAX

JavaScript war jedoch nicht immer so beliebt bei Entwicklern, denn früher war man der Meinung die Funktionalität der Sprache wäre von leistungsfähigeren Technologien überholt worden und man sollte ohnehin den Server die ganze Arbeit machen lassen.

Vor einigen Jahren bekam JavaScript eine kleine Erweiterung, mit der es möglich war den nervigen Datenverkehr in den Hintergrund zu verlegen und Inhalte dynamisch von dem Server laden, ohne das der Client etwas davon mitbekommt. Diese Technologie heißt Asynchronous JavaScript and XML (AJAX) und ist der wichtigste Bestandteil von JavaScript und der Diplomarbeit.

[12]

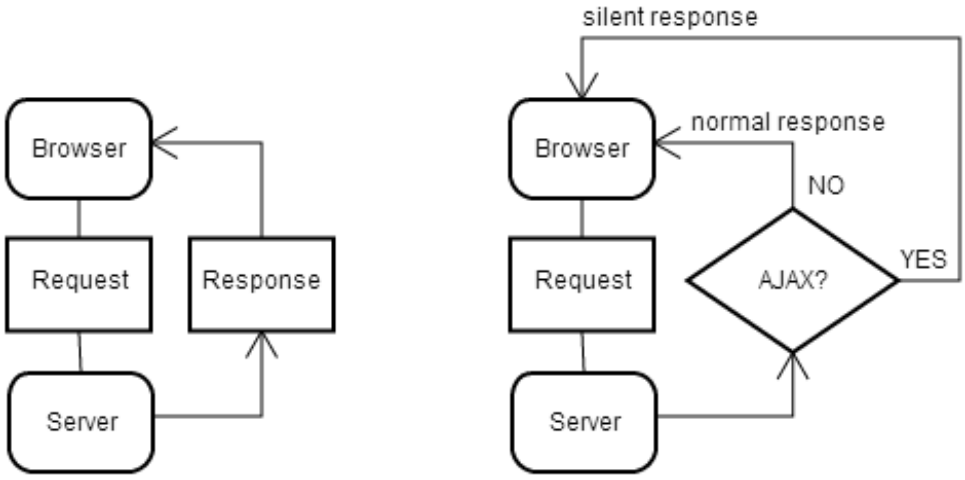

Abb. 29: Flussdiagramme HTTP-Request (li.) & kombiniert mit AJAX-Request (re.)

Einen AJAX-Request zu schicken ist an und für sich keine große Kunst mehr heutzutage. Mit Bibliotheken wie jQuery benötigt man keine 4 Zeilen und hat schon mehrere Dateien asynchron nachgeladen. Nachfolgend dasselbe Skript in nativen JavaScript (Code 11) und mit der Bibliothek jQuery (Code 12).

```
var obj = document.querySelector("button");
obj.addEventListener("click", send, false);
function send() {
    var obj = document.getElementById("intro");
    var xmlhttp;
    if (window.XMLHttpRequest)
        // code for IE7+, Firefox, Chrome, Opera, Safari
        xmlhttp=new XMLHttpRequest();
    else // code for IE6, IE5
        xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
    xmlhttp.open("GET","file.php",false);
    xmlhttp.send();
    obj.innerHTML=xmlhttp.responseText;
}
```

Code 11: AJAX-Request mit JavaScript[14]

```
$(document).ready(function() {
    $(document).on('click','button',function() {
        $.ajax({
            url: 'file.php',
            success: function(msg){
                $('#intro').html(msg);
            }
        });
    });
});
```

Code 12: jQuery AJAX-Request[15]

Spätestens hier hat jeder jQuery-Skeptiker aufgegeben. Der Code in Code 12 ist transparenter, atomarer und ist für jeden Browser der Gleiche.

Man kann mit AJAX auch Datenpakete hin- und her schicken, dafür hat sich eine spezielle Notationsform als sehr Hilfreich erwiesen, die man JSON (JavaScript Object Notation) nennt. JSON ist eine sehr effiziente Form, Daten eines Objektes darzustellen, wie man in Code 13 eindeutig sehen kann.

```
{
    "ID": 1,
    "name": "John Doe",
    "time": "08:00:00",
    "data": [120,302,85]
}
```

Code 13: Ein JSON-Objekt[16]

Das JSON Objekt aus Code 13 wird von zwei geschwungenen `{}`-Klammern umschlossen und enthält 4 Eigenschaften, denen entweder ein String-Wert zugewiesen wird oder ein Array, das von zwei eckigen `[]`-Klammern umschlossen wird.

10.2.4. PHP

Entspricht das Dokument den Vorstellungen des Entwicklers/Kunden, gilt es als nächsten Schritt eine solide Logik zu schaffen. Dafür wird die Serverseitige Skriptsprache PHP (Hypertext Preprocessor) herangezogen. Um PHP auf einem Webserver lauffähig zu machen muss zuerst eine Installation vorgenommen werden, falls das noch nicht geschehen ist. Im Kapitel „XAMPP“ wird näher darauf eingegangen.

10.2.4.1. PHP-Parser

Mit PHP ist es möglich webbasierte Dokumente, wie zum Beispiel die eingangs erwähnten HTML-Dokumente dynamischer zu gestalten. Man kann sie hierfür ganz leicht in Sinnabschnitte teilen und danach wieder an die richtigen Stellen einfügen. Das geht mit

[14] http://www.w3schools.com/ajax/ajax_xmlhttprequest_send.asp (20.02.2014)
[15] https://api.jquery.com/jQuery.ajax/ (20.02.2014)
[16] http://www.json.org/ (20.02.2014)

der Funktion `include('file.php')` und dient der Vermeidung von Redundanzen, also doppelten Anweisungen. Dabei spielt PHP eine wichtige Rolle. Mit dieser serverseitigen Skriptsprache ist es möglich das Request-Response Modell um 2 wesentliche Bestandteile zu erweitern, wie man in Abb. 30 sieht:

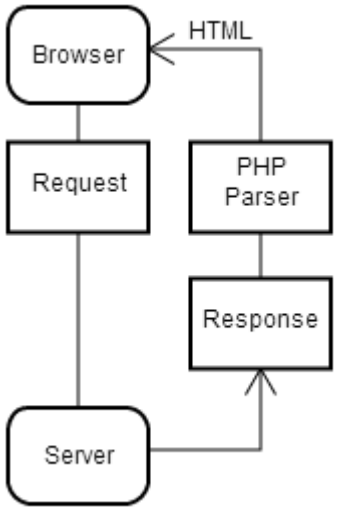

Abb. 30: Request-Response Modell mit PHP-Parser

Beschreibung zu Abb. 30:

- Der Client schickt über den Browser einen Request an den Server, ein Dokument zu suchen.
- Handelt es sich bei dem angeforderten Dokument um ein PHP-Skript, erkennt das der Server an der Dateiendung .php und liefert die Seite nicht direkt an den Client aus, sondern übergibt sie dem PHP-Parser.
- Der PHP-Parser interpretiert die PHP-Befehle und versucht ein HTML-Dokument zu erzeugen.
- Das erzeugte HTML-Dokument wird an den Browser zurückgesendet. Es heißt zwar noch genauso, wie das angeforderte Dokument, enthält aber keinerlei PHP-Code mehr.

Das ermöglicht die Erstellung von HTML-Dokumenten, deren Inhalte stets aktuell sind, da sie dynamisch mit Daten aus Dokumenten oder einer Datenbank gefüttert werden können. PHP ist also für die Logik hinter webbasierten Anwendungen verantwortlich, denn ohne diese Skriptsprache wären Webseiten nicht das, was sie heute sind.

[Mau12]

10.2.4.2. Syntax[17]

In PHP kann man Variablen definieren, die neben numerischen auch textuelle oder eine Kombination aus diesen annehmen können. Ebenso ist es möglich den Variablen einen

[17] http://php.net/manual/de/language.types.array.php (29.01.2014)

boolschen (`true`/`false`) Wert oder keinen (`null`) Wert zuzuweisen. Neben diesen einfachen Variablen ist es erlaubt, eine Art Liste oder Karte zu definieren. Das nennt man Array und ist die bevorzugte Art eine größere Menge Daten zu speichern.

```php
<?php
    $number = 147; // Zahl
    $letters = "Hallo"; // Wort
    $collection = array("Hallo",42,"Welt",false); // Array
?>
```

Code 14: Variablendeklarationen

Die Syntax von PHP ist sehr an die von C angelehnt, jedoch ist es in PHP nicht notwendig den Datentyp von Variablen zu definieren. Der PHP-Parser sucht sich automatisch den passenden. Man definiert also einen Integer-Wert genau gleich, wie einen vom Typ String, was man in Code 14 gut erkennen kann.

Eine wichtige Funktion für die Entwicklung in PHP ist `var_dump()`. Diese gibt nämlich jede Art von Variablen aus und sagt einem, welchen Datentyp eine jede besitzt. Zusammen mit dem HTML-Tag `<pre></pre>` ergeben die Beiden eine mächtige Kombination, mit der schnell & übersichtlich entwickelt werden kann. Das Beispiel aus Code 15 zeigt eine typische Fehlerbehebungs-Situation, in der man den Inhalt eines Arrays ausgibt.

```php
<?php
    $collection = array("Hallo",42,"Welt",false);
    var_dump($collection);
    /* output:
    array(4) {
      [0]=>
      string(5) "Hallo"
      [1]=>
      int(42)
      [2]=>
      string(4) "Welt"
      [3]=>
      bool(false)
    } */
?>
```

Code 15: Hallo Welt Beispiel in PHP

10.2.4.3. Formulare mit PHP

```
<?php
    if(isset($_POST) && count($_POST) > 1) {
        echo '<pre class="text-left">';
        var_dump($_POST);
        echo '</pre>';
    }
?>
<h1>PHP Formular</h1>
<form action="<?=$_SERVER["PHP_SELF"];?>" method="POST">
    <section>
        <label for="name">Ihr Name</label>
        <input type="text" id="name" name="name" class="form-control" />
    </section>
    <section>
        <label for="name">Ihr Geschlecht</label><br />
        <div class="btn-group" data-toggle="buttons">
            <label class="btn btn-default">
                <input type="radio" name="gender" value="m">
M&auml;nnlich
            </label>
            <label class="btn btn-default">
                <input type="radio" name="gender" value="f"> Weiblich
            </label>
        </div>
    </section>
    <section>
        <label for="tel">Ihre Telefonnummer</label>
        <input type="tel" id="tel" name="tel" class="form-control" />
    </section>
    <input type="submit" class="btn btn-primary" value="OK">
</form>
```

Code 16: HTML-Ausschnitt mit eingebetteter PHP-Verarbeitung

Das in Code 16 gezeigte Dokument stellt eine `input`-Box zur Eingabe des Names, zwei `radio`-Boxen, zur Auswahl des Geschlechts, sowie eine weitere `input`-Box zur Eingabe der Telefonnummer des Besuchers. Klickt man dann auf den `submit`-Button, werden die Daten mittels `$_POST`-Array an das Skript (dasselbe Dokument) geschickt und und dort auf vorhandensein mit der Funktion `isset()` geprüft. Danach wird geprüft, ob das Array `$_POST` mehr als 0 Stellen hat, ist das der Fall wird es schön Formatiert im `<pre></pre>`-Element ausgegeben.

Wie man in Abb. 31 erkennt, zeigt das Array `$_POST` an der Stelle `name` den eingegebenen Wert an. Man könnte genauso gut `$_POST["name"]` schreiben, es hätte dasselbe Ergebnis zur Folge. Ebenso gibt es zwei `input`-Felder vom Typ `option`, welche einen unterschiedlichen Wert haben, nämlich `f` für female und `m` für male.

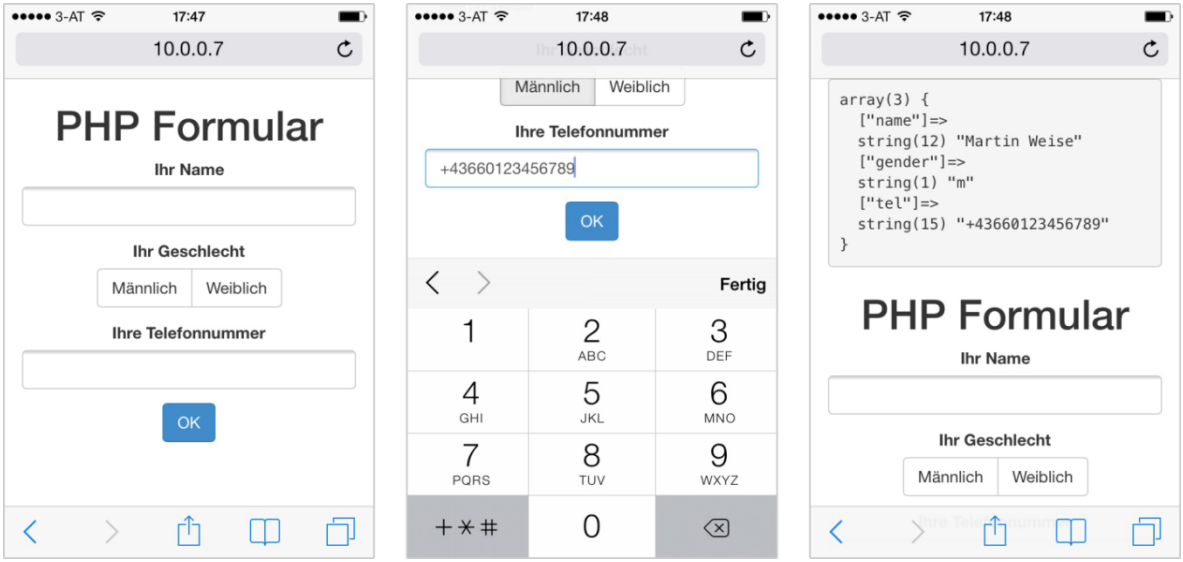

Abb. 31: Formularverarbeitung mit PHP

10.2.5. MySQL

Wie im vorangegangenen Punkt PHP bereits erwähnt, wirdeine Datenbank benötigt. Dabei gilt es zu beachten, dass die Technologie Relationale Datenbanken beherrscht und unter einer Lizenz veröffentlicht wurde, die sich mit den Anforderungen deckt. Der Begriff „Relation“ ist fest definiert und beschreibt die Verbindung zweier Tabellen mittels eindeutigen Werten. Um sich davon ein besseres Bild machen zu können, wird das nachfolgend näher erläutert.

[Mau12]

10.2.5.1. Begriffe

Eine Tabelle besitzt im Allgemeinen mindestens eine Reihe und mindestens 1 Spalte. Die Spalte einer Tabelle wird in der Fachliteratur als Attribut und eine Zeile wird als Tupel bezeichnet. Um einen Tupel von einem anderen unterscheiden zu können, hat jede Tabelle ein oder mehrere identifizierende Attribute, die als Schlüsselattribute (Keys) bezeichnet werden. Identifiziert soein Attribut den Tupel eindeutig, also ist das Attribut einmalig vorhanden, so wird das Schlüsselattribut als Primärschlüssel bezeichnet, der wichtigste Key. Im Normalfall ist das eine fortlaufende, ganze, natürliche Zahl und trägt den Spaltennamen `` `ID` ``.

[Mau12]

Dient dieser jedoch lediglich als Verweis zu einem Primärschlüssel einer anderen Tabelle, wird dieses Schlüsselattribut Fremdschlüssel, also Foreign Key genannt. Der Primärschlüssel ist hingegen in jeder Tabelle vorhanden, daher kann man später sehr gut und vor allem einfach Verbindungen zwischen den Tabellen herstellen. Um eine Tabelle von einer anderen, sogar einer strukturell gleichen unterscheiden zu können, weist man jeder Tabelle eine Relationsvariable, also eine Tabellenbezeichnung zu. Diese enthält

üblicherweise ein Präfix, das die Projektzugehörigkeit symbolisieren soll. In Abb. 32 ist das „pf_“ (in der Beschriftung) eine Abkürzung für „präfix“.

[Mau12]

ID	notice	color
1	Rot	DD1E2F
2	Geld	EBB035
3	Grün	218559
4	Blau	336699
5	Hellblau	06A2CB
6	Grau	D0C6B1
7	Braun	834C24

Abb. 32: Tabelle pf_colors

Ein Relationales Datenbankmanagementsystem (RDBMS) wird durch die Structured Query Language (SQL) beschrieben und bedient. Dabei gibt es viele verschiedene Anbieter, verschiedenster Lösungen.

[Mau12]

10.2.5.2. Syntax[18]

Wie man in Code 17 erkennt, schreibt man SQL-Schlüsselwörter, wie `CREATE`, `TABLE` oder die Datentypen `INT`, `VARCHAR`, `DOUBLE` allesamt in Großbuchstaben, was der besseren Lesbarkeit dient und den Code wesentlich übersichtlicher macht. Zudem rückt man nach jedem Anweisungsblock ein.

```
USE `test`;
CREATE TABLE IF NOT EXISTS `pf_colors` (
    `ID` int(11) NOT NULL AUTO_INCREMENT,
    `notice` varchar(100) NOT NULL,
    `color` char(6) NOT NULL,
    PRIMARY KEY (`ID`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 AUTO_INCREMENT=9;
INSERT INTO `pf_colors` (`ID`, `notice`, `color`) VALUES
    (1, 'Rot', 'DD1E2F'),
    (2, 'Gelb', 'EBB035'),
    (3, 'Grün', '218559'),
    (4, 'Blau', '336699'),
    (6, 'Hellblau', '06A2CB'),
    (7, 'Grau', 'D0C6B1'),
    (8, 'Braun', '834C24');
```

Code 17: Erstellen der Tabellenstruktur aus Abb. 32

[18] http://www.1keydata.com/de/sql/ (27.01.2014)

Die Syntax macht sofort einen sehr strukturierten Eindruck. Die Anweisungsblöcke lassen sich wie ein Text lesen.

`USE db_name` sagt dem MySQL-Interpreter, dass er die Datenbank `` `test` `` für alle folgenden Statements verwenden soll. Die Datenbank bleibt bis zum Ende der Sitzung oder bis ein weiteres `USE`-Statement folgt gespeichert.[19]

`CREATE TABLE [IF NOT EXISTS] tbl_name`erzeugt eine neue Tabelle mit dem Namen `tbl_name`. Dieses Statement wird auch ausgeführt, wenn derzeit keine Standardtabelle ausgewählt wurde. Man muss nur die Datenbank, dem Tabellennamen voranstellen und die beiden mit einem Punkt verbinden.

Der `PRIMARY KEY` identifiziert einen Datensatz eindeutig. Dieser kann aus einer- oder mehreren Spalten Bestehen. In diesem Fall wird nur eine Spalte in den Primärschlüssel einbezogen, nämlich die Spalte `ID`.

Als Engine für die Tabelle wird die `InnoDB` Storage Engine verwendet, weil die Funktionen nach dem ACID[20]-Modell arbeiten. Als Charakter Set wurde UTF-8 gewählt, weil dieses jedes Zeichen in dem international weit verbreiteten Set, dem Unicode, speichern kann.[21]

Generell gilt es, so viel, wie nur irgend möglich über die Datenbank zu regeln. Ein Query ist im eigenen Netzwerk mit einer durchschnittlichen Ausführzeit von etwa 250 Millisekunden um ein beachtliches Stück schneller als PHP. Daher versucht man alle Abfragen, Filterungen und Sortierungen mit SQL zu lösen und lediglich 1 einzigen Query an die Datenbank zu schicken. Dadurch spart man sehr viel Ladezeit und das ganze System wird wesentlich beschleunigt.

Man überlässt der Datenbank die Berechnung der gefundenen Datensätze, somit muss man nurnoch die gefundenen Zahlen richtig ausgeben, das spart eine Menge Zeit in der Ausführung. Es wird für die drei Suchkriterien ein eigenes Attribut generiert und ein Tupel erzeugt, der für die drei Kriterien einen ganzzahligen Wert annimmt, nämlich die Anzahl der gefundenen Datensätze.

10.2.5.3. Einsatz

Für die Diplomarbeit wurde ein lokaler Webserver verwendet, der einen vorkonfigurierten MySQL-Server beeinhaltet. Dieses Softwarepaket nennt sich XAMPP und ist untennäher erklärt.

10.2.6. XAMPP

Im Normalfall wird der Server irgendwo außerhalb der Räumlichkeiten des Benutzers stehen und dort unter optimalen Bedingungen (gekühlte Raumtemperatur, niedrige

[19] http://dev.mysql.com/doc/refman/5.0/en/use.html (13.02.2014)
[20] ACID = Atomicity, Consistency, Isolation, Durability
[21] http://dev.mysql.com/doc/refman/5.1/en/create-table.html (13.02.2014)

Luftfeuchte, Schalldämmung, etc.) betrieben. Der Browser läuft daheim auf dem Notebook, Tablet oder Smartphone des Benutzers. Diese Aufstellung ermöglicht eine Spezialisierung auf den jeweiligen Verwendungszweck und trägt somit zu einer höheren Effizienz maßgeblich bei.

10.2.6.1. Lokaler Server

Mit dem Serverpaket XAMPP (Software, die auf einem beliebigen Betriebssystem (X) Apache, MySQL, Perl, PHP enthält) ist es möglich, mit nahezu keinem Konfigurationsaufwand einen kleinen Server auf dem Heimrechner zu installieren. Zudem sind alle genannten Komponenten Open-Source, was bedeutet, dass deren Nutzung zu welchem Zweck auch immer, stets kostenlos bleibt. Es wird die Serverkomponente auf den Client geholt und es ist möglich unter realen Bedingungen zu testen. Damit bieten sich dem Benutzer unter anderem zwei neue Möglichkeiten auf:

- Es ist nun möglich einen Server im lokalen Netzwerk zu betreiben und so eine Umgebung zu schaffen, mit der man (vereinfacht) lokal Web-Applikationen testen kann, die sonst einen Server außerhalb der Räumlichkeiten des Benutzers benötigt hätten.
- Ein mögliches Anwendungsgebiet ist die Speicherung von persönlichen Dokumenten, Bildern oder Musik auf dem so geschaffenen Server, die man im lokalen Netzwerk aufrufen kann.

Diese Aspekte sind insbesondere für Personen der Entwicklungsbranche sehr hilfreich und verbessern den Workflow (siehe Abschnitt 10.2.7) durch minimierten Einrichtungsaufwand, praktisch keine Kosten für einen Server & Software und die optimalen Deploymentbedingungen, also die Vorraussetzungen, dass Software schnell am Rechner des Benutzers verfügbar und verwendbar ist.

Die 3 wichtigsten Module aus der Serverumgebung für die Diplomarbeit sind der Apache Server, PHP und die MySQL Datenbank. Das Modul Apache lässt sich mit diesem Abschnitt schon sehr gut beschreiben, die Module PHP und MySQL sind in den Abschnitten 10.2.4 und 10.2.5 näher erklärt.

[Sei06]

10.2.6.2. Apache[22]

Der Begriff „Apache“ bezeichnet lediglich die Apache Software Foundation, welche unter anderem den wohl am Häufigsten verwendeten Webserver[23] der Welt entwickelt, den Apache HTTP Server.

[22] http://httpd.apache.org/ABOUT_APACHE.html (08.02.2014)
[23] http://www.netcraft.com/survey/ (08.02.2014)

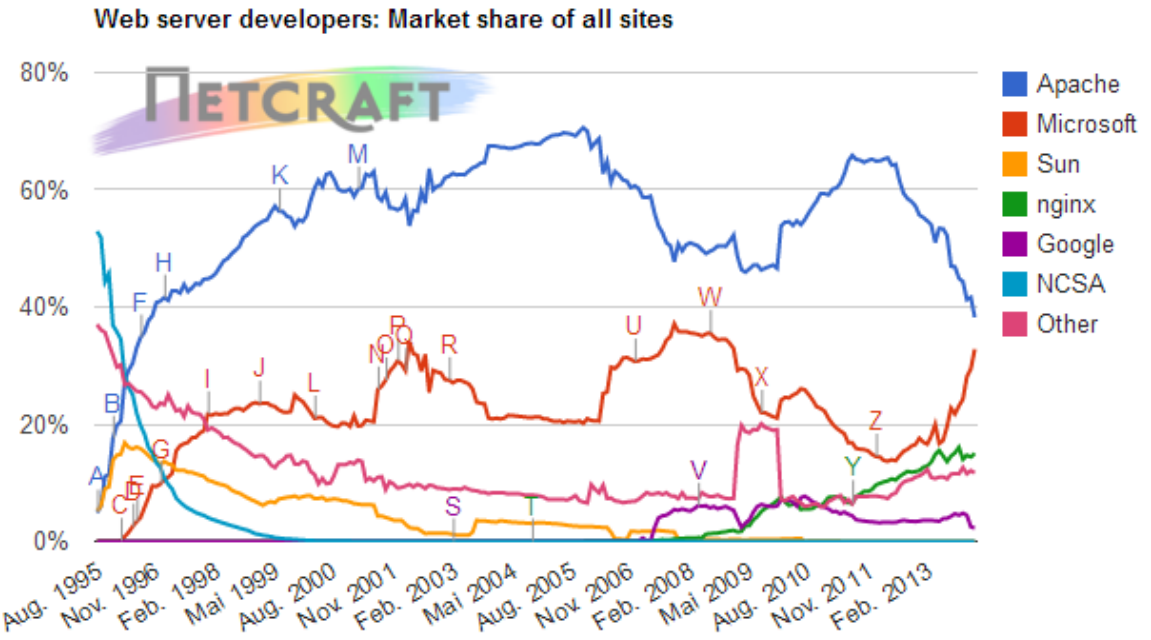

Abb. 33: Anteil der Webserver-Technologien

Das Ziel ist schnell erklärt: Die gemeinsame Entwicklung eines robusten, frei erhältlichen HTTP (Web) Servers, der viele Funktionen aufweisen kann. Das Projekt wird von vielen Freiwilligen, die überall auf der Welt verteilt sitzen verwaltet und geführt. Sie benützen das Internet und das Web für Kommunikation, Planung, Entwicklung des Servers und der Dokumentation.

Jeder kann bei der Weiterentwicklung helfen. Je mehr man geholfen hat, desto mehr Rechte bekommt man bei der Entwicklung. Das Entscheidungsrecht ist dabei dem Komitee vorbehalten, das entscheidet, ob eine Änderung an dem Originalcode vorgenommen wird, oder nicht. Das Original-Projektverzeichnisist nämlich für jeden Entwickler schreibgeschützt, nur das Komitee kann Änderungen bewilligen. Dafür wird an alle Komitee-Mitglieder eine E-Mail geschrieben. Geben drei ihr „+1“ und keines ein „-1“, dann wird die Änderung beim nächsten, größeren Patch wirksam gemacht.

Die eigentlichen Änderungen geschehen also erst einmal auf den lokalen Maschinen der Entwickler, bevor sie in der Mailingliste diskutiert oder vorgeschlagen werden. Dem Komitee kann man als Entwickler nach einer Zeit von sechs Monaten beitreten, wenn ein Mitglied des Komitees einen Nominiert.

Die Apache Software ist ohne Einschränkungen irgendeiner Art kostenlos und frei verfügbar für jeden Menschen der Zugang zum Internet hat. Die Software ist und bleibt auf unbestimmte Zeit kostenlos, weil die Apache Software Foundation der Meinung ist, dass Ein Produkt, wie es ein Webserver ist, stets kostenlos verfügbar sein muss, um das Internet frei von „Besitzern“ zu halten, welche das Internet gerne als „ihr Eigentum“ ansehen wollen würden. Die Protokolle des Internets (zB: HTTP) sollen stets frei von

Besitzern gehalten werden, deshalb muss man jedem einen Zugang zu einem Teil des Protokolls geben, mithilfe eines Webservers.

10.2.6.3. phpMyAdmin

Mit dem Softwaretool phpMyAdmin ist es für Web-Administratoren leichter geworden, Zugang zu dem MySQL-Dienst eines Web-Servers zu erhalten und zu verwalten. Das Tool ist ebenfalls kostenlos und komplett in PHP geschrieben. In neueren Versionen ist besteht es mittlerweile aus PHP (ca. 79.8%), JavaScript (19.6%) und anderen Sprachen (0.6%)[24]. Die Hauptgründe, um phpMyAdmin zu nutzen sind die große Community, die große Stabilität und das funktionelle User-Interface, mit dem man Tabellen, Datenbanken, Spalten, Tupel, Relationen, Indizes, Benutzer, Rechte, usw. verwalten kann. Außerdem ist das User-Interfacehervorragend geeignet, um SQL-Statements auszuführen und diese zu entwickeln/testen.

Mit dem Serverpaket XAMPP ist bereits ein User im Werkszustand festgelegt, der alle Rechte am MySQL-Server genießt, der *root*-User. Dieser kommt im Werkszustand ohne Passwort, sodass man im Browser nur *http://localhost/phpmyadmin* aufrufen muss, um sofort Zugriff auf oben genannte Funktionen zu haben.

10.2.7. Workflow (in Anlehnung an [Zil13])

Als Entwickler von Software hat man als Freiberuflicher oder als Agentur einen bestimmten Ablauf, den man durchführt, um ein Softwareprojekt möglichst schnell und kostengünstig im Rahmen der veranschlagten Kosten abzuwickeln. Dabei hat jeder Freelancer oder jede Agentur ihre eigenen Workflow, der meist auf Erfahrungswerten, Ressourcen und dem Kundenfeld basiert.

10.2.7.1. Mobile First

Betrachtet man ein Projekt auf seine grundlegenden Anforderungen, hat es sich für Web-Entwickler als sinnvoll erwiesen, mit einer mobilen Version der Web-Applikation zu starten. Diese Vorgehensweise wird seit einigen Jahren vom amerikanischen Webdesigner Luke Wroblewski verbreitet, der sie als den „Mobile First“-Ansatz bezeichnet. Der soll sicherstellen, dass in der Planungsphase der Fokus auf den wesentlichen Inhalten gelenkt wird. Weil dieser Ansatz in Symbiose mit dem Inhalt lebt, wird auch oft von einem „Content First“-Ansatz gesprochen. Das soll verdeutlichen, dass man sich verstärkt mit den Inhaltsfragen beschäftigen soll.

Wenn man nur einen Blick auf die Verkaufszahlen und Wachstumsraten von Smartphones im Vergleich zu denen von PCs wirft, sieht man, warum es zwingend notwendig ist, Web-Applikationen „Mobile First“ anzugehen. Im Jahr 2011 wurde weltweit mehr Smartphones als PCs verkauft. In absoluten Zahlen gesprochen heißt das 488 Millionen (+63% zu 2010) verkaufte Smartphones und 415 Millionen (+15% zu 2010) PCs. Da die Diplomarbeit aber

[24] https://github.com/phpmyadmin/phpmyadmin (Stand: 08.02.2014)

für Tablets und Desktops konzipiert wird, sind die Verkaufszahlen von Tablets von großem Interesse. Diese wurden 2011 nämlich 27 Millionen (+274% zu 2010) Mal[25] verkauft.

10.2.7.2. Planungsphase

Von dem „Mobile First“-Ansatz profitieren auch Endgeräte mit höherer Auflösung als ein Smartphone, wie ein Tablet und der altbekannte PC. Denn durch eine klare Struktur und durch übersichtliche Elemente, sowie einem Fokus auf elementare Funktionen erreicht man auch bei Nutzern von großen Bildschirmen gewisse Sympathie. Vorteile sind zum Beispiel:

- Schnellere Ladezeiten durch weniger „Füll-Inhalt“
- Mehr Klarheit durch sinnvolle Strukturierung
- Besser durchdachtes Gesamtkonzept

10.2.7.3. Designphase

Geht es in die Designphase, in der man versucht ein visuelles Konzept seiner Web-Applikation auszuarbeiten, hat es sich für jedes Projekt bewährt, dem „Mobile First“ Ansatz nur eingeschränkt zu folgen.

Abb. 34: Accessability von Elementen bei Touch-Oberflächen

Von daher wurde hier auf das bekannte 960px-Grid gesetzt, welches den Inhalt der Webseite auf 960 Pixel reduziert. Sinn dahinter ist die einfache und unkomplizierte Unterteilung, weil sich 960 sehr leicht durch eine beliebige natürliche Zahl teilen lässt. Folgende Fragen können bei diesem Prozess dabei sehr hilfreich sein:

- Wo platziere ich welche Inhalte?
- Wie soll der Gesamteindruck der Applikation sein?
- Wie soll die farbliche Gestaltung sein?
- Wie ist die Accessability (siehe Abb. 34) der Elemente mit Touch-Eingabegeräten?

[25] http://canalys.com/newsroom/smart-phones-overtake-client-pcs-2011

Hat man diese Fragen geklärt, gilt es von den vorliegenden Entwürfen die restlichen Entwürfe zu besprechen. Man sollte sich nun Gedanken über kleinere und größere Bildschirme machen. Dabei darf man sich nicht dazu verleiten lassen, die Desktop Ansicht zu vernachlässigen. Es kann nämlich durchaus auch vorkommen, dass dann ein sog. Content-out entsteht und wichtiger, interessanter Inhalt fehlt und dem Nutzer dann eine langweilige Web-Applikation geboten wird.

Das Design darf nicht zu kurz kommen. Das Zusammenspiel von Bildern, Texturen, Layout und Schriftstil sind auf mobilen und stationären Endgeräten genauso unverzichtbar – es kommt auf die Dosierung an.

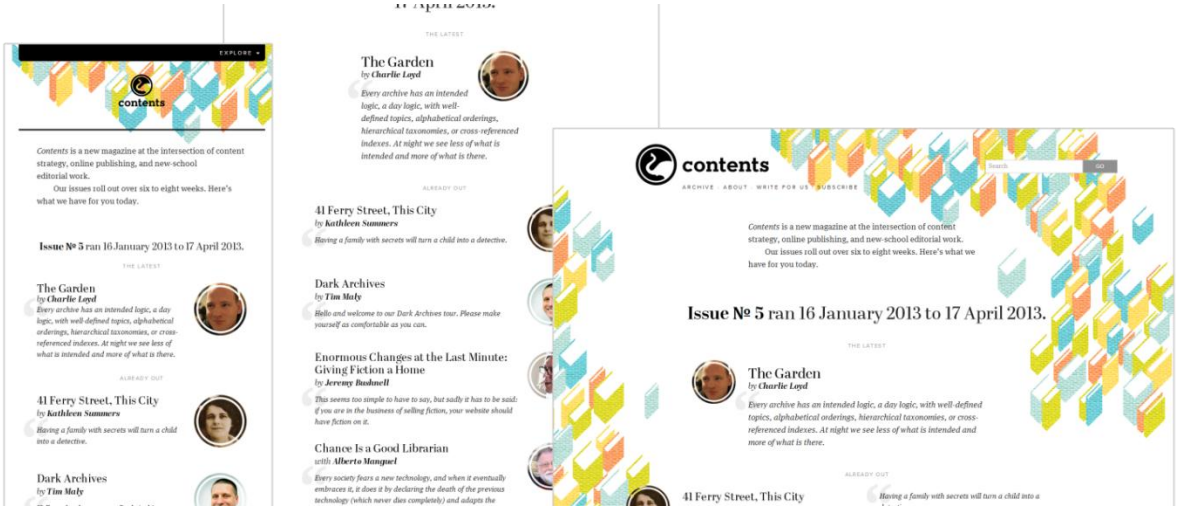
contents
Contents is a new magazine at the intersection of content strategy, online publishing, and new-school editorial work.
Our issues roll out over six to eight weeks. Here's what we have for you today.
Issue № 5 ran 16 January 2013 to 17 April 2013.
The Garden
by Charlie Loyd
Every archive has an intended logic, a day logic, with well-defined topics, alphabetical orderings, hierarchical taxonomies, or cross-referenced indexes. At night we see less of what is intended and more of what is there.
41 Ferry Street, This City
by Kathleen Summers
Having a family with secrets will turn a child into a detective.
Dark Archives
by Tim Maly
Hello and welcome to our Dark Archives tour. Please make yourself as comfortable as you can.
Enormous Changes at the Last Minute: Giving Fiction a Home
by Jeremy Bushnell
Chance Is a Good Librarian
with Alberto Manguel

Abb. 35: Gelungenes Design über verschiedene Bildschirmauflösungen

Alles in allem ist es wichtig, flexibel zu bleiben und sich bei dem Entwerfen von Skizzen nicht in Details zu verrennen. Die Leute von Contents Magazine[26] haben das sehr vorbildlich umgesetzt, wie man in Abb. 35 sehr gut erkennen kann. Es wird nämlich im Laufe der Gestaltung immer wieder vorkommen, dass ein Konzept, das für einen Anwendungsbereich interessant ist, dann rückwirkend auf andere ebenso interessant wirkt und dann auch in Bereiche übernommen wird, die eigentlich als Abgeschlossen galten.

10.2.7.4. Entwicklungsphase

Hier hat der „Mobile First“-Ansatz wieder die volle Gültigkeit, die zuvor festgehaltene Anordnung der Elemente wird in den Fokus genommen. Es gilt mögliche Konzeptfehler mit dem Design abzuklären und es möglichst vielen Auflösungen verschiedener Endgeräte recht zu machen.

[Zil13]

[26] http://contentsmagazine.com/

11. Programmieren (Umsetzung)

11.1. Entwicklungsumgebung

Für das rasche Umsetzen einer komplexen Anwendung wie der Diplomarbeit, ist es notwendig flexibel und effizient arbeiten zu können. Dafür stehen Entwicklern diverse Werkzeuge zur Verfügung, wie der Datei-Explorer, ein Textverarbeitungsprogramm und der Browser für zB. Datenbankverwaltung.

Je größer die Anzahl der verwendeten Entwicklerwerkzeuge, desto schwieriger wird es bei der Sache, also bei der Arbeit, zu bleiben. Muss man ständig zwischen Datei-Explorer und Textverarbeitungsprogramm hin-und-herschalten verliert man schnell das Interesse zB. noch eine weitere Stunde zu arbeiten. Also setzt man auf sog. integrierte Entwicklungsumgebungen (engl. IDE = Integrated Developemnt Environment). Diese bringen zumindest die vorhin genannten Funktionen mit, die von uns verwendete Software PhpStorm.

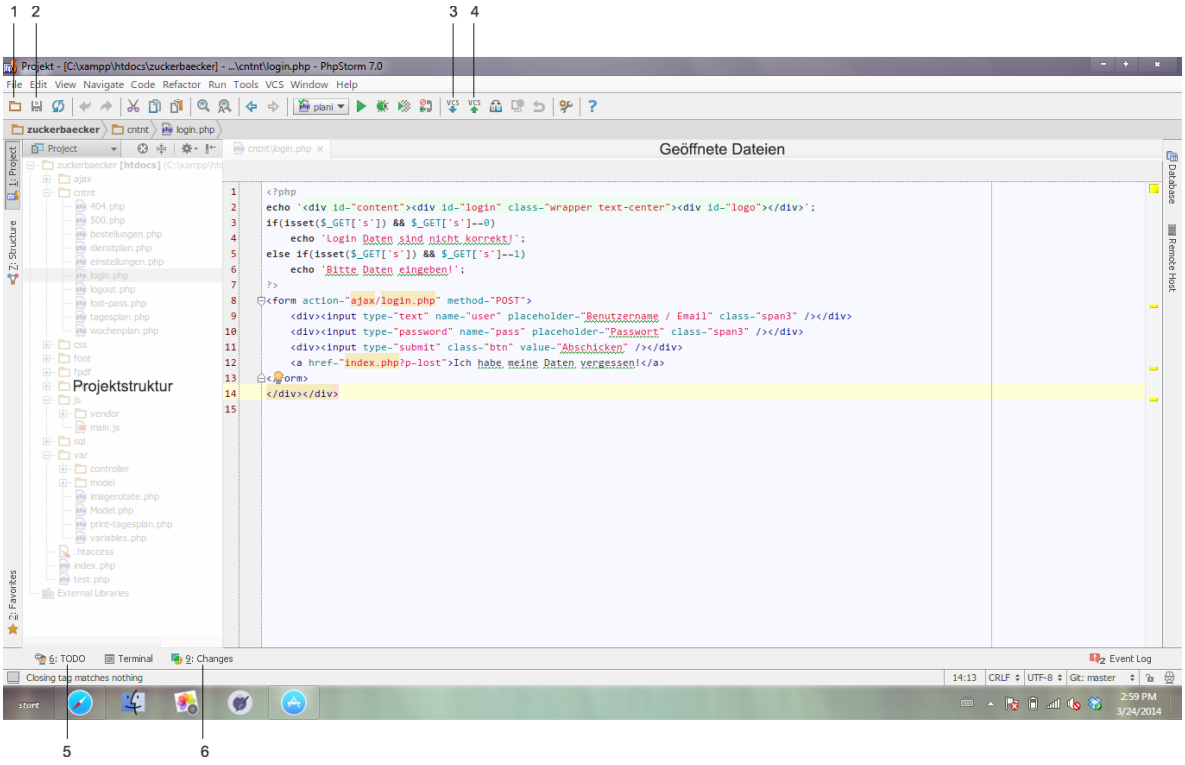

Abb. 36: PhpStorm 7.0 Entwicklungsumgebung

11.1.1. Funktionen

Die verwendeten Kernfunktionen sind in Abb. 36 markiert. Man findet einen komplettes Dateiverwaltungssystem vor, mit dem man zB: öffnen (1) und speichern (2) kann. Zudem kann man externe Versionskontrollsysteme verwenden. Um eine Version auf einen externen Server hochzuladen muss man nur VCS-Push (4) drücken, um seine Version auf

dem vorher konfigurierten Server hochzuladen. Steigt man frisch in das Programm ein, oder möchte Änderungen von anderen Entwicklern herunterladen, drückt man VCS-Pull (3), um diesen Vorgang einzuleiten. Die Änderungen zur aktuellen Versionen werden dann im Changes-Log (6) angezeigt, wenn man möchte sogar Zeile für Zeile. Möchte man vor dem Verlassen der IDE noch eine Anmerkung für das nächste Mal hinterlassen, kommentiert man mit `// TODO: Anmerkung`, sofort scheint im TODO-Log (5) die entsprechende Zeile mit Dateinamen auf und man sieht die Anmerkung.

11.2. Datei-Struktur

Das Model View Controller (MVC) Modell ist in der Programmierung besonders beleibt, weil es Präsentations- und Logikschicht klar voneinander trennt. Diese Separierung ist wichtig, daher wurde die Diplomarbeit nach diesem Aspekt gestaltet. Hinzu kommen noch andere Verzeichnisse, die einen eigenen Sinn verfolgen.

11.2.1. ajax-Verzeichnis

Dieses Verzeichnis beinhaltet jede Datei, die über einen AJAX-Request angefordert wird. Das sind im Normalfall PHP-Skripte, welche für die Auswertung von HTML-Formularen zuständig sind.

11.2.2. cntnt-Verzeichnis

In diesem Verzeichnis befinden sich das Layout des Wochenplans, Dienstplans, Tagesplans, der Bestellungen, Einstellungen und weiterer Seiten.

11.2.3. css-Verzeichnis

Hier befinden sich die CSS-Dateien, wie `bootstrap.min.css`, `normalize.css` (setzt alle HTML-Elemente auf normale Werte für alle Browser) und `page.css` (Stylesheet).

11.2.4. font-Verzeichnis

Wie in Kapitel 10.2.2.1 erwähnt, ist mit CSS3 viel Neues dazugekommen. So kann man Beispielsweise jede beliebige Schriftart innerhalb der Web-Applikation verwenden, auch wenn man diese nicht mal installiert hat.

Die Schriftart Open-Sans[27] ist unter der Apache-License 2.0 veröffentlich worden (d.h. darf man mit dem Projekt Geld verdienen, auch wenn die Schriftart nicht von einem stammt) und unterstützt alle erdenklichen Sonderzeichen und Schrifttypen. Daher wurde u.a. aus diesem Grund diese Schriftart gewählt.

[27] http://www.google.com/fonts/specimen/Open+Sans (02.03.2014)

11.2.5. fpdf-Verzeichnis[28]

In diesem kommen alle Dateien, welche für die fpdf-Klasse benötigt werden, unter. Das beinhaltet auch Dokumentation, Tutorials und die Character-Maps für verschiedene unterstützte Schriftarten.

11.2.6. img-Verzeichnis

Hier sind alle Bilder der Web-Applikation untergebracht. Es sind mehrere Unterverzeichnisse vorzufinden.

11.2.6.1. background-Unterverzeichnis

Alle Hintergrundbilder der Web-Applikation.

11.2.6.2. img-Unterverzeichnis

Alle Bilder im PNG-Format, wie das Herz für den Tagesplan, usw.

11.2.6.3. u-Unterverzeichnis

Alle von Benutzer (=User, kurz: u) hochgeladenen Bilder, wie zB durch das Formular in Abb. 22.

11.2.7. js-Verzeichnis

Das beinhaltet die Haupt-JavaScript Datei, main.js und ein Unterverzeichnis.

11.2.7.1. vendor-Unterverzeichnis

Hier werden alle JavaScript Dateien von externen Anbietern (engl. vendor). Die Skripte `bootstrap.min.js`, `jquery.min.js` (Bibliothek), `jquery-mobile.custom.min.js` (Taphold-Funktion, man könnte auch Bibliotheken wie hammer.js[29] verwenden), `jquery.noui-slider.min.js` (Der Schieberegler der Mitarbeiter-Arbeitszeit), `jquery.stayInWebApp.min.js` (wichtig um eine am Homescreen abgelegte Web-Applikation etwas mehr wie eine native App wirken zu lassen), `jquery-ui.min.js` (Draggable, Droppable), `jquery-ui-touch-punch.min.js` (wichtig, weil jquery-UI alleine nicht auf mobilen Geräten unterstützt wird).

11.2.8. sql-Verzeichnis

Beinhaltet ein Backup der Datenbank im SQL-Format.

11.2.9. var-Verzeichnis

Dieses beinhaltet den Druckbaren Tagesplan (`print-tagesplan.php`), der mit der fpdf-Klasse arbeitet, sowie die Projektvariablen (`variables.php`), die für die Template-

[28] http://www.fpdf.org/ (02.03.2014)
[29] https://github.com/EightMedia/hammer.js/ (05.03.2014)

Klasse wichtig sind und die Datei, welche die aufgenommenen Bilder in die richtige Rotation versetzt, sodass sie immer aufrecht sind und im quadratischen-Verhältnis stehen als auch 2 Unterverzeichnisse.

11.2.9.1. controller-Unterverzeichnis (Controller)

Beinhaltet die Template-, Main- und SQL-Klasse.

11.2.9.2. model-Unterverzeichnis (Model)

Beinhaltet das Header-Modell, welches auf jeder Seite zu finden ist.

11.3. Front-End[30]

Das Front-End ist jener Teil der Applikation, der für das präsentieren von Daten, Texten, Bildern, Videos, also jeder Information, zuständig ist. Komponenten des Front-Ends sind das Design, die User-Experience, die Usability, und viele weitere.

11.3.1. Template

Das `index.php` ist das Grundskript, in welches alle Inhalte dynamisch über PHP und AJAX geladen werden. Hinter dieser Überlegung steckt sehr viel Erfahrung und Wissen über die Technologie.

```php
<?php
    session_start();
    error_reporting(E_ALL);

    require_once 'var/variables.php';
    require_once 'var/model/Header.php';
    require_once 'var/model/Button.php';
    require_once 'var/model/ButtonGroup.php';
    require_once 'var/controller/Template.php';
    require_once 'var/controller/SQL.php';
    require_once 'var/controller/Login.php';

    $db = newSQL();
    $login = newLogin();
    $template = new Template();
?>
```

Code 18: Einbinden aller MVC-Module in das index.php-Skript

Durchsucht man mit einem Browser ein Verzeichnis auf einem Webserver, sucht dieser immer nach einer Datei, die `index` heißt, die Dateiendung ist erst einmal egal. Das stellt für das wichtigste Dokument sicher, dass es auf jeden Fall zuerst ausgeführt wird. Die Dateiendung `.php` deklariert die Datei als ein PHP-fähiges Skript, welches durch den PHP-Parser interpretiert werden muss, bevor der Request beendet werden kann und ein Response folgt.

[30] http://whatis.techtarget.com/definition/front-end (17.02.2014)

Diese zwei Überlegungen sind wichtig, denn nun kann eine Sitzung mit `session_start()` gestartet werden, anschließend werden alle etwaigen Fehlermeldungen mit `error_reporting()` für die Anzeige aktiviert und alle MVC-Skripte eingebunden (dazu später).

Nun werden Instanzen der Klassen `SQL`, `Login` und `Template` erstellt, es wird vorerst zur Klasse `Template` näher Bezug genommen.

```
<title>Planify &bull; <?=$template->title;?></title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<meta name="viewport" content="width=device-width, initial-scale=1.0,
maximum-scale=1.0, user-scalable=no"/>
<meta name="HandheldFriendly" content="true"/>
<meta name="apple-touch-fullscreen" content="yes"/>
<meta name="apple-mobile-web-app-capable" content="yes"/>
<meta name="apple-mobile-web-app-status-bar-style" content="black"/>
<meta name="apple-mobile-web-app-title" content="Planify">
```

Code 19: title-Tag und meta-Angaben im head-Bereich des index.php

In Code 19 wird klar, dass auch meta-Angaben in einem HTML-Dokument nicht fehlen sollten, denn jede dieser Angaben ist wichtig für eine bestimmte Anforderung. Die erste Anweisung sagt dem Browser, dass es sich um ein `text/html`-basiertes HTML-Dokument handelt, dessen Bytes mit `utf-8` umzuwandeln sind.[31]

Die Anweisung `viewport` bestimmt wie das Dokument auf mobilen Endgeräten dargestellt werden soll. Hier wird dem Browser gesagt, er soll die Breite des Dokuments auf die maximale Bildschirmbreite des Geräts (`device-width`) einstellen. Zudem soll das Dokument in 1-facher Vergrößerung dargestellt werden und eine Vergrößerung durch den Benutzer nicht zulässig sein (`user-scalable=no`). Diese Einstellung ist ziemlich populär und bei manchen Webentwicklern verpönt, weil sie den Benutzer bevormundet und ihm die Freiheit nimmt das Dokument nach seinem Belieben zu skalieren.[32]

Spezieller Erklärung bedarf es noch den `apple-*` Attributen. Mit der Anweisung `apple-touch-fullscreen` ist es möglich die Web-Applikation, vorausgesetzt man betrachtet das Dokument mit einem Apple-Mobilgerät und startet es vom Homescreen, ohne Browsersteuerungen zu starten. Diese Notation wurde durch `apple-web-app-capable` ersetzt. Die Web-Applikation ähnelt dann sehr einer nativen iOS-App.

Die Anweisung `apple-mobile-web-app-status-bar-style` bestimmt die Farbe des Statusleiste, wenn man die Web-Applikation über den Homescreen im Vollbildmodus startet. Der Wert wurde auf `black` geändert, um sicherzustellen, dass die Farbe der Statusleiste auch weiterhin schwarz bleibt.

[31] http://de.selfhtml.org/html/kopfdaten/meta.htm (17.02.2014)
[32] http://html5-mobile.de/blog/meta-viewport-fuer-mobile-anpassen (17.02.2014)

Wenn man die Web-Applikation zum Homescreen hinzufügt soll der voreingestellte Name Planify sein, das stellt `apple-mobile-web-app-title` sicher.[33]

```
<link rel="icon" type="image/x-icon" href="img/png/favicon.png"/>
<link rel="shortcut icon"href="img/touch/apple-touch-icon.png">
<link rel="apple-touch-icon-precomposed"
href="img/png/favicon72x72.png"/>
<link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="css/jquery.gridster.css">
<link rel="stylesheet" type="text/css" href="css/normalize.css">
<link rel="stylesheet" type="text/css" href="css/page.css">
<link rel="stylesheet" type="text/css" href="css/jquery.nouislider.css">
```

Code 20: Verlinkte externe Ressourcen des index.php

Um externe Ressourcen in Verbindung mit dem `index.php` bringen zu können, gibt man eine Ressource mithilfe des `<link />`-Tags in das `index.php` an, diese sind selbstschließend, benötigen also keinen Closing-Tag. Dabei spielt das Attribut `rel` eine wichtige Rolle. Es gibt an, welchen Zweck die verlinkte Ressource für das Dokument hat, das Attribut `type` hingegen gibt über den Typ der verlinken Ressource Auskunft.[34]

So lässt sich zum Beispiel das Icon einer Web-Applikation sehr leicht festlegen, wie man in Code 20 (Anweisung 1-3) sehen kann. Spezieller wird es da schon mit der Vergabe des Icons der Web-Applikation, wenn man bei (mobilen) Geräten der Firma Apple Inc. die Web-Applikation zum Homescreen hinzufügen möchte. Hierfür benötigt man die Apple-spezifische Relationsangabe `apple-touch-icon-precomposed`. Der Name der Datei muss auch mit den Richtlinien von Apple übereinstimmen, für das iPad ist das Dateinamensuffix `72x72`[35].

Cascading Stylesheets werden über die Relationsangabe `stylesheet` und dem Dateityp `text/css` angegeben.

```
<link rel="icon" type="image/x-icon" href="img/png/favicon.png"/>
<link rel="shortcut icon" href="img/touch/apple-touch-icon.png">
<link rel="apple-touch-icon-precomposed"
href="img/png/favicon72x72.png"/>
<link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="css/jquery.gridster.css">
<link rel="stylesheet" type="text/css" href="css/normalize.css">
<link rel="stylesheet" type="text/css" href="css/page.css">
<link rel="stylesheet" type="text/css" href="css/jquery.nouislider.css">
```

Code 21: Verlinkte JavaScript-Skripte des index.php

Mit Hilfe des `<script></script>`-Tags kann man in das HTML Dokument Clientseitige Skriptsprachen, wie zB: JavaScript, einbinden oder auf eine externe Ressource verweisen. Für die Diplomarbeit waren die Bibliothek jQuery, jQuery Mobile, jQuery UI, ein

[33] https://developer.apple.com/library/safari/documentation/AppleApplications/Reference/SafariHTMLRef/Articles/MetaTags.html (17.02.2014)
[34] http://www.w3schools.com/tags/tag_link.asp (19.02.2014)
[35] http://iconhandbook.co.uk/reference/examples/favicons/ (19.02.2014)

kleines Script stayInWebApp-Skript, welches verhindern soll, dass eine Homescreen-Web-Applikation den Standardbrowser öffnen soll, und diverse slider-Skripts, sowie dem `main.js`-Skript, welches die Web-Applikation verbessert. Später findet man noch die Skripte `bootstrap.js`, `modernizr.js` und `jquery-ui-touch-punch.js`, welche ebenfalls zu einer besseren Usability beitragen und nur aus Ladezeittechnischen Gründen nicht im `head`-Bereich des HTML-Dokuments stehen.[36]

```
if(isset($_SESSION['id']) && !empty($_SESSION['id'])){
    // START NAVIGATION
    echo '<div class="menu toggle-hide" data-role="header">
    <div id="logo"></div><ul class="nav nav-tabs nav-stacked">';
    // ABFRAGE NACH SEITENRECHTEN
    $query='SELECT `wp`, `dp`, `pr`, `ta`, `ei` FROM `pf_login` WHERE `ID` = :id';
    $db->query($query,array(
        ':id' => mysql_real_escape_string($_SESSION['id'])
    ));
    foreach($template->pages as $i => $item) {
        if($item['nav']===true && $db->count > 0 && isset($db->row[0][$item['permission']])
        && $db->row[0][$item['permission']] == 1)
            echo'<li><a href="index.php?p='.strtolower($item['title']).'">'
            .$item['title'].'</a></li>';
    }
    echo'</ul><div class="text-center"><a href="cntnt/logout.php">Logout</a></div></div>';
    // ENDE NAVIGATION
    echo'<div class="display scrollable toggle-show animation" data-role="content">
    <div class="text-center"><noscript>Bitte JavaScript wieder einschalten!</noscript>
    </div>';
}
// UNABHÄNGIG VON LOGIN, WIRD DER INHALT GELADEN
$template->loadPage();
// ENDE
if(isset($_SESSION['id'])&&!empty($_SESSION['id']))
    echo'</div>';
```

Code 22: Template-System des index.php

Ist der Benutzer (siehe Code 22) über das Login-Formular im System angemeldet, wird das richtige Template im body-Bereich ausgegeben, welches ein Logo und eine Navigation vorsieht. Dazu wird in der Datenbank nach den Berechtigungen der 5 Hauptmodule gefragt und im Array `$db->row[0]` gespeichert.

Nun wird für alle im System angelegten Seiten selektiert, ob diese überhaupt in der Navigation erscheinen sollen [`$item['nav'] === true`, man prüft bei boolschen Werten und `null` immer auf den Datentyp und nicht auf deren Wert (entspr. ==)]. In der Datenbank wird die Erlaubnis ein Modul verwenden zu können mit dem Wert 1 festgelegt. Es wird also auf Wertegleichheit mit 1 geprüft, bei einer Übereinstimmung wird der entsprechende Listenpunkt ausgegeben. Dem folgt ein Logout-Link.

Der aber wohl interessanteste Punkt ist die Einbindung des Inhalts der aufgerufenen Seite, dieser wird über die Klasse `Template()` (siehe 11.4.1) mit der Funktion `loadPage();` geladen (siehe 11.4.1.5).

[36] http://www.w3schools.com/TAGS/tag_script.asp (19.02.2014)

11.3.2. main.js-Datei

```
$(function(){//DOM Ready
    $.stayInWebApp('a');
    $(document).on('taphold','#tagesplan li.item:not(.ui-sortable-
helper):not(.lieferant)', tp_change_value);
    $(document).on('click','#tagesplan li.item:not(.ui-sortable-
helper):not(.lieferant) button.btn.btn-primary', tp_change_value_click);
    $(document).on('click','#wochenplan
li.item:not(.lieferant):not(button):not(select)',wp_sel);
    [...]
});
```

Code 23: Teil der Event Handler

Am Anfang wird definiert (s. Code 23), beim Klicken auf welches Element verhindert werden soll, dass die Web-Applikation verlassen werden soll. Dafür wird die Funktion `stayInWebApp` der `jquery.stayInWebApp.min.js` verwendet und auf das Element `a` angewandt, welches im Normalfall dazu führt, dass die Web-Applikation verlassen wird.

Die Aktion `taphold` wird durch die `jquery-mobile-xxx.min.js` bereitgestellt.

```
if(!("url"in window)&&("webkitURL"in window)) {
    window.URL = window.webkitURL;
}
function takePicture(event) {
    $(this).parent().ajaxForm({
        success: $.proxy(function showResponse(responseText,...) {
            $(this).parent().parent().find('img').data('url',responseText);// IMG
        },this),
        statusCode: {
            400:function() {
                $('#msg').show().html('<h4>Fehler!</h4>[...]').fadeOut(8000);
            }
        }
    });
    // AUTO-PREVIEW
    if(event.target.files.length == 1
    && event.target.files[0].type.indexOf("image/") == 0)

$(this).parent().parent().find('.image').attr("src",URL.createObjectURL(event.tar
get.files[0]));
    $(this).parent().submit();
}
```

Code 24: BilduploadSkript aus 2 Teilen bestehend

Das in Code 24 zu sehende Fragment ist wichtig für den automatischen Upload von Bildern bei den Mitarbeiter-Einstellungen. Die erste `if`-Anweisung prüft, ob der Browser Webkit unterstützt. Tut dieser das, muss die `window.URL` Variable auf die des Webkits abgeglichen werden.[37]

[37] https://developer.mozilla.org/en-US/docs/Web/API/Window.URL (03.03.2014)

Die Funktion `takepicture()` benötigt nämlich den korrekten Pfad für die Zwischenspeicherung des Abbilds und macht nichts anderes, als das hochzuladende Bild lokal anzuzeigen, ohne, dass es noch gespeichert wurde.

Anschließend wird das Formular mit AJAX abgeschickt und das Bild somit über das im Markup definierte Skript `ajax/upload_file.php` hochgeladen. Dieses prüft die Datei auf Größe (<5 MB) und auf den Dateityp. Jeder Fehler gibt einen HTTP-Status von 400 (Bad Request) zurück, der heißt, dass etwas schief gegangen ist. Im Normalfall wird aber HTTP-Status 200 (OK) zurückgegeben, was mit der Funktion `success` einhergeht. Mit `data()` wird das Bild mit dem Dateinamen des erfolgreich hochgeladenen Bildes beschrieben. Das ist für die Datenbank sehr wichtig, falls sich der Benutzer dazu entscheiden sollte auf „Speichern“ zu drücken.

Hat `event.target.files` eine Länge von 1 und ist die in `event.target.files` enthaltene Datei vom Typ `image/*`, wird das `src`-Attribut des Bildes anstatt mit dem sonst üblichen Verweis auf eine Datei, nun mit einem Objekt, das aus diesem Zwischengespeicherten Bild generiert wurde, befüllt.

11.3.3. Komponenten

Man kann die Web-Applikation in die Teile: Wochenplan, Dienstplan, Tagesplan, Produktion, Einstellungen, Leiharbeiterauswahl und Spezialbestellungen einteilen. Im Folgenden werden alle diese Komponenten im Detail erläutert, deren Code erklärt und bei Bedarf ein Screenshot angefügt.

11.3.3.1. Wochenplan

Im Wochenplan werden 10 JavaScript-Funktionen zur Abdeckung des geforderten Produktumfangs benötigt. Im Folgenden wird eine Auswahl der wichtigsten Funktionen beschrieben.

```
function wp_sel(){
	$('li.item[data-id="'+$(this).data('id')+'"]')
	.toggleClass('selected');
}
```

Code 25: JavaScript wp_sel Funktion

In Code 25 wird die Funktion `wp_sel()` dargestellt. Wie sich vermuten lässt, macht diese nichts anderes, als allen Elementen des Wochenplans mit der Klasse `item` und dem Attribut `data-id`, dessen Wert mit dem angeklickten Element übereinstimmen muss, eine Klasse zuzuweisen oder zu entfernen.

Die Klasse `selected` ist etwas ganz spezielles. Sie ändert den Hintergrund des Elements zu einem 45°-gestreiften Fortschrittsbalken. Dieses Muster wird dann auch noch von links nach rechts bewegt und simuliert so den „Fortschritt“. Die neue Hintergrundfarbe hängt von der alten ab.

```
@keyframes progress-bar-stripes {
    from {
        background-position: 40px 0;
    }
    to {
        background-position: 0 0;
    }
}

#wochenplan li.item.selected {
    background-image: linear-gradient(
        45deg,
        rgba(255, 255, 255, 0.15) 25%,
        transparent 25%,
        transparent 50%,
        rgba(255, 255, 255, 0.15) 50%,
        rgba(255, 255, 255, 0.15) 75%,
        transparent 75%,
        transparent
    );
    background-size: 40px 40px;
    animation: progress-bar-stripes 2s linear infinite;
}
```

Code 26: CSS-Deklaration der Klasse selected

Mit der `@keyframes` Regel kann man, wie in Code 26 zu sehen ist, Animationen definieren.[38] In diesem Fall wird der simulierte Fortschrittsbalken (der mit 40x40 Pixel bemessen ist) um seine Breite, von 40 Pixeln in Richtung 0 verschoben, diese Animation dauert 2 Sekunden und wird unendlich oft wiederholt.

```
function wp_drag_single(event,ui){
    $(this).addClass('single');
    $(this).on('mouseup',function(){
    if($(this).css('left')=='auto'|| $(this).css('top')=='auto')
            $(this).removeClass('single');
    });
}
```

Code 27: JavaScript wp_drag_single Funktion

Drückt man lange die linke Maustaste (oder mit dem Finger), wird die Funktion `wp_drag_single()`, die in Code 27 abgebildet ist, aufgerufen. Diese fügt an das angeklickte Element die Klasse single an und einen weiteren Eventhandler, der beim Loslassen der Taste aktiviert wird. Diese Funktion wird aber erst dann Interessant, wenn man die Funktion wp_sort, die in Code 28 abgebildet ist, hinzuzieht.

Diese sucht nach allen Elementen des Wochenplans mit der Klasse `item` und demselben Wert des Attributs `data-id`, wie den des gerade bewegenden Elements. Die gefundenen Elemente werden alle mit dem bewegten Element mitverschoben. Die Positionen berechnen sich wie folgt:

[38] http://www.w3schools.com/cssref/css3_pr_animation-keyframes.asp (10.03.2014)

$$pos_x = draggable_{offsetLeft} - item_{index} * draggable_{width} - item_{index} * 4$$
$$pos_y = draggable_{offsetTop} + offset * 45 + offset * 4$$

Die Berechnung von pos_x aus erfolgt von dem Abstand nach Links des bewegenden Elements minus dem `index` des aktuellen Elements (Mo: 0, Di: 1; etc.) mal der Breite des bewegenden Elements. Der Index ist üblicherweise negativ, daher gilt: je höher der `index`, desto weiter rechts ist das Element platziert. Dazu wird (üblicherweise) ein Abstand von $item_{index}$ mal 4 addiert, der aus Formatierungstechnischen Gründen notwendig ist.

```
function wp_sort(event,ui) {
    if(!$(ui.item).hasClass('single')) { // MEHRERE ELEMENTE
        $('.tr').eq(0).find('ul').each(function() {
            offset=0;
            $('li.item[data-id="'+$(ui.item).data('id')+'"]').each(function() {
                index=$(ui.item).parent().index()-$(this).parent().index();
                $(this).css({
                    position:'absolute',
                    width: $(this).parent().width(),
                    left: parseInt($(ui.item).css('left'))-
                    index*parseInt($(ui.item).css('width'))-index*4,
                    'z-index': 9000,
                    top: parseInt($(ui.item).css('top'))+offset*45+offset*4
                });
                offset++;
            });
        });
    } else { // EINZELNES ELEMENT
        [...]
    }
}
```

Code 28: JavaScript wp_sort Funktion

In Code 29 wird abermals jede Liste durchsucht und jedes Element der Liste dem Array data hinzugefügt. Wichtig und Interessant sind die Daten `stepID`, `orderID`, `date` und `index`. Ist das Array fertig befüllt, wird ein AJAX-Request an das PHP-Skript `wochenplan_update_item.php` geschickt, welcher die neuen Positionen aller Elemente in die Datenbank schreibt.

Anschließend werden die Klassen `sender`, `m` und `single` von den entsprechenden Elementen entfernt und die neuen Zeiten berechnet.

```
function wp_sort_stop(event, ui) {
    [...]
    $('.tr').eq(0).find('ul').each(function() {
        $(this).find('li.item[data-
id="'+$(ui.item).data('id')+'"]').each(function() {
            $(this).css('width', $(this).parent().width());
            data.push({
                "date": $(this).parent().data('date'),
                "stepid": $(this).data('stepid'),
                "orderid": $(this).data('id'),
                "index": koeff
            });
        });
    });
    $.ajax({ // ALLE REFRESHEN
        type:"POST",
        url:"ajax/wochenplan_update_item.php",
        data:{'json': data },
        success: $.proxy(function(message){
            console.log(message);
        },this)
    });
    $('.sender').removeClass('sender');
    $('.m').removeClass('m');
    $('.single').removeClass('single');
    wp_calc_time();
}
```

Code 29: JavaScript wp_sort_stop Funktion

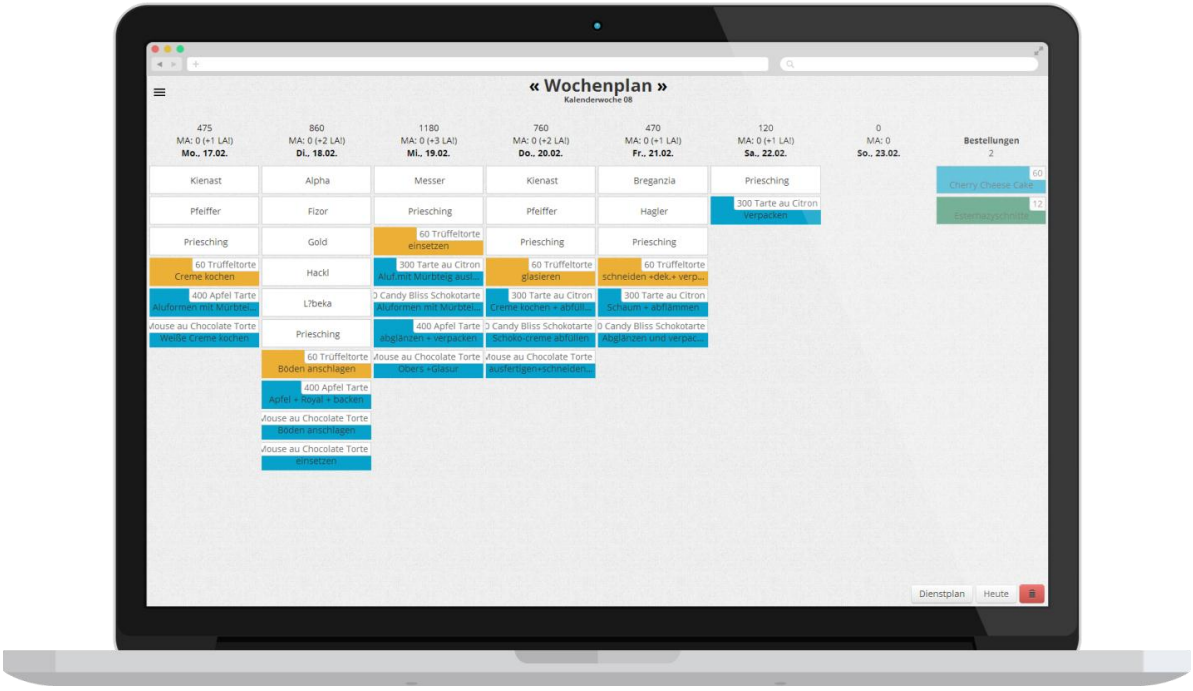

Abb. 37: Wochenplan

11.3.3.2. Dienstplan

Das Front-End des Dienstplans ist nicht ganz so kompliziert, wie das des Wochenplans. Dennoch werden für die definierte Funktionalität des Dienstplans ca. 9 Funktionen benötigt.

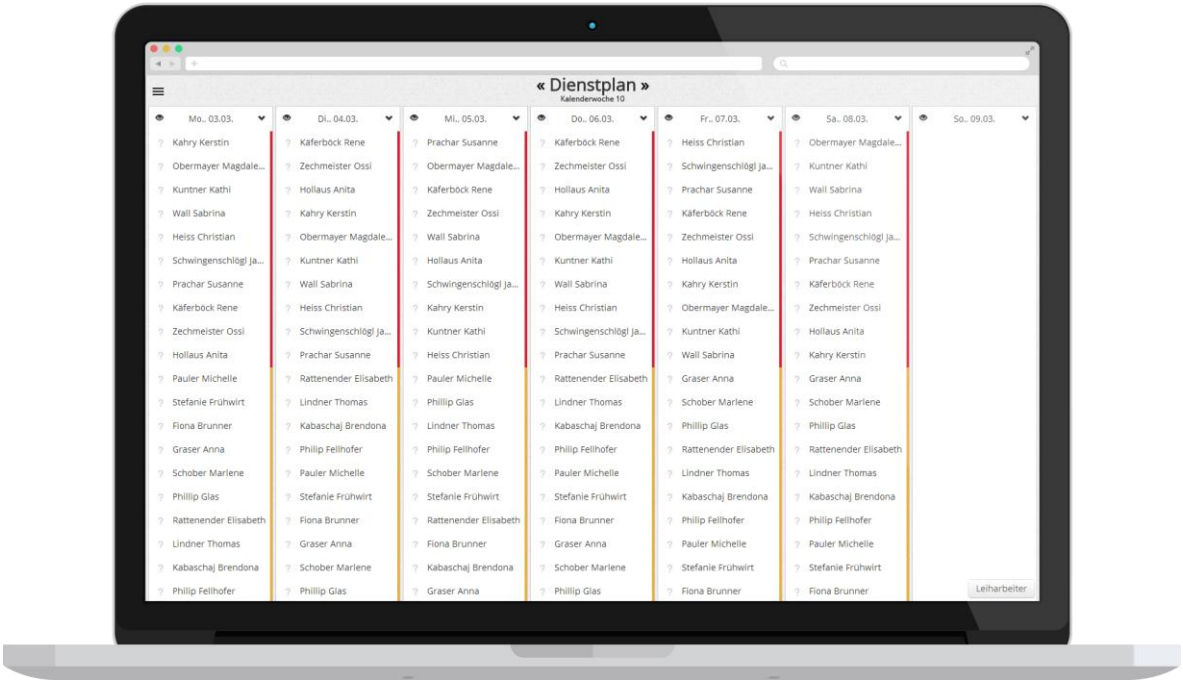

Abb. 38: Dienstplan

```
function dp_show_employees() {
    if($('.sender').length > 0)
        return false;
    var element;
    if(event.type=="click") {
        element = $(this).parent().parent();
        if($(this).hasClass("rotate-180")){ // AUSGEKLAPPT
            $(this).removeClass("rotate-180");
            $('.tr').eq(0).find('ul').eq(element.index()).html('');
        return;// AUSKLAPPEN VERHINDERN
        } else {
            $(this).addClass("rotate-180");
        }
    } else {
        element = $(this);
    }
    element.find("i.pull-right").addClass("rotate-180");
    $.ajax({
        type: "POST",
        url: "ajax/roster_display_employees.php",
        data: {'date': element.data('date')},
        beforeSend: function() {
            $(document).find('ul.itemlist').eq(element.index()).show()
            .html('<li class="text-center"><img src="img/loading.gif"
alt="..." /></li>');
        },
        success: $.proxy(function(msg) {
            var json = JSON.parse(msg);
            $(document).find('ul.itemlist').eq(element.index()).html('');
            for(var i=0; i<json.length; i++) {
                $(document).find('ul.itemlist').eq(
element.index()).append([...]);
            }
            if(json.length <= 0)
            element.find('i.icon-chevron-down').removeClass('rotate-
180');
            // DRAGGABLE MACHEN
            [...]
        },this)
    });
}
```

Code 30: JavaScript dp_show_employees Funktion

Die Funktion liefert in Code 30 automatisch ein Objekt event mit. Für die Funktion interessant ist die Eigenschaft `event.type`, weil diese Auskunft über die Aktivierungsart gibt. Wird das Zirkumflex angeklickt und es zeigt die verfügbaren Mitarbeiter, werden diese versteckt und das Zirkumflex wird wieder normal gedreht. Werden diese nicht gezeigt, so wird es gedreht.

In jedem Fall werden die noch fehlenden Mitarbeiter geladen und in dem Platzhalter angezeigt.

```
function dp_drop(event,ui) {
    var data = {
        date: $(this).data('date'),
        eid: parseInt($(ui.draggable).data('eid'))|| null,
        id: parseInt($(ui.draggable).data('id'))|| null,
        coords: [$(ui.draggable).parent().index(),
$(ui.draggable).index()]
    }
    if($(this).find('li.item[data-eid="'+$(ui.draggable).data('eid')+'"]
    [data-pid="'+$(ui.draggable).data('pid')+'"]').length ==0) {
        if($(ui.draggable).parent().parent().index()==0) // NEU
            $.ajax({
                type: "POST",
                url: "ajax/roster_add_employees.php",
                data: { json: data }
            });
        else // NUR UPDATE
            $.ajax({
                type:"POST",
                url:"ajax/roster_update_employees.php",
                data:{ json: data }
            });
    $(ui.draggable).css({
        left:0,
        top:0
    }).appendTo(
        $(this).finish().css('background','#fff').effect('highlight',{
            color: $(ui.draggable).css('border-right-color')
        },1000)
    );
    }
        dp_calc_time();
}
```

Code 31: JavaScript dp_drop Funktion

Zuerst wird in Code 31 das Datenobjekt gefüllt. Danach wird übeprüft, ob am Zielort bereits ein Mitarbeiter mit denselben Informationen existiert. Wird der Mitarbeiter zu einem Posten neu zugeteilt, wird das PHP-Skript `roster_add_employees.php` mittels AJAX aufgerufen. Andernfalls wird die Zuteilung des Mitarbeiters über einen AJAX-Request an `roster_update_employees.php` geschickt.

Anschließend werden noch die Stilinformationen des Elements zurückgesetzt und eine Hintergrund-Animation am Ziel-Container vollführt.

11.3.3.3. Tagesplan

Der Tagesplan benötigt 6 verschiedene Funktionen. Übliche Funktionen sind die Verschiebe-aktionen der Mitarbeiter zwischen den Posten und der Liste der Nichteingeteilten & der Produktschritte zwischen den Posten.

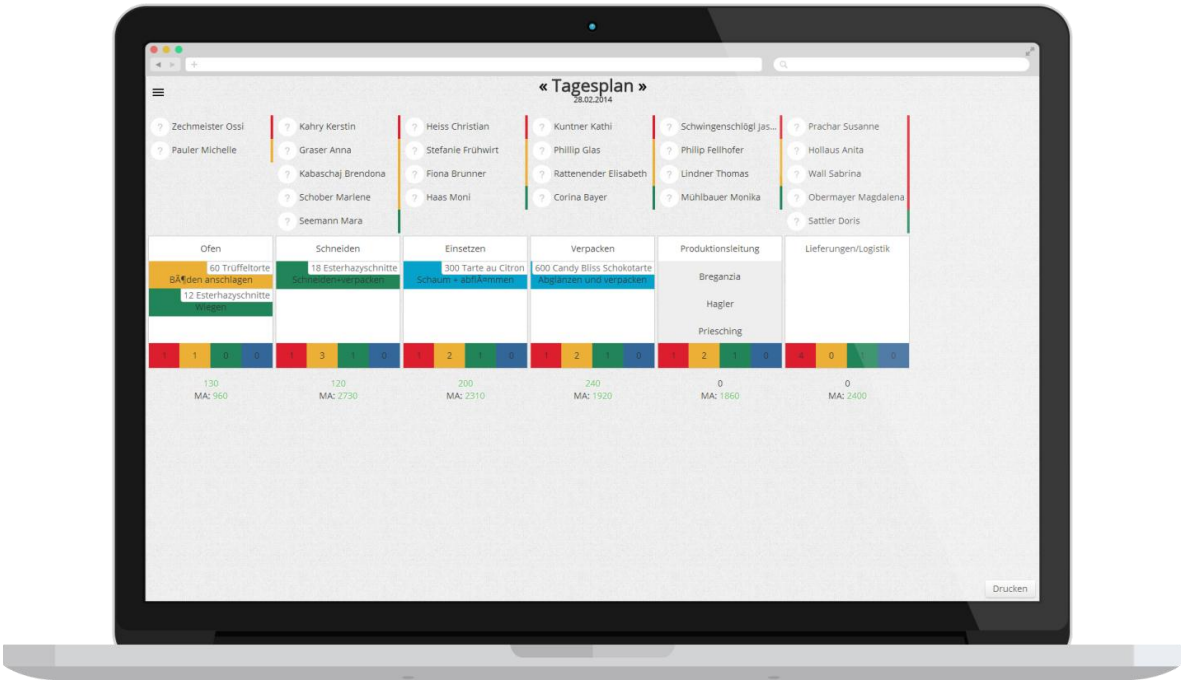

Abb. 39: Tagesplan

Der Ablauf in Code 32 ist wie folgt: Es werden die Zahlen aller Mitarbeiter (die bunten Vierecke unten an jedem Posten) auf 0 gesetzt. Nun wird einfach für jedes Element die passende Mitarbeiterzahl um 1 erhöht am betreffenden Posten.

```
function tp_employee_count(){
    $('#tagesplan ul.posten ul.employees li').html('0');
    $('#tagesplan li.eitem').each(function(){
var e = $('#tagesplan ul.posten').eq($(this).parent().index())
.find('ul.employees li').eq((parseInt($(this).data('pid'))||1)-1);
e.html((parseInt(e.html())||0)+1);
});
}
```

Code 32: JavaScript tp_employee_count Funktion

Das in Code 33 zu findende Skript behandelt die Aktion, die man einleitet, wenn man ein verschiebbares Element im Tagesplan verschiebt und dann loslässt. Diese Elemente sind sowieso nur auf den dafür vorgesehenen Flächen fallen zu lassen, andersrum würden sie einfach nur zurückgesetzt werden.

Die folgende `if`-`else`-Bedingung regelt die Animation, beim loslassen. So wird, wenn das Element auf einem Posten fallen gelassen wird, einfach die Farbe jedes Postens auf Weiß zurückgesetzt und dann auf dem aktuellen Posten die Rahmenfarbe des Elements kurz animiert.

```
function tp_drop(event,ui) {
    var e = $(this).find('li ul.employees
li').eq($(ui.draggable).data("pid")-1);
    var color = $(ui.draggable).css('border-right-color');
    var temp =($(ui.draggable).data('pid')==4?1:0);
    var data = {
        "eid": $(ui.draggable).data('eid'),
        "pid": $(this).index(),
        "date": $('#tagesplan').data('date'),
        "temp": temp
    };
    if(!$(ui.draggable).parent().is(':last-child')) {
        $('.tr').eq(1).find('ul:not(:last)').css('background','#fff');
        $(this).finish().effect('highlight',{color:color},1000);
    } else {
        // HOVER EFFEKT ZURÜCKSETZEN + LOSLASSEFFEKT EINLEITEN
        $(this).css('background','#fff').finish().effect('highlight',{
        color:color},1000);
    }
    $.ajax({
        type:"POST",
        url:"ajax/tagesplan_add_employee_to_posten.php",
        data:{"json": data }
    });
    $(ui.draggable).appendTo($('.tr:first').find('ul')
    .eq($(this).index())).css({
        left:0,
        top:0
    });
    tp_calc_time();
    tp_employee_count();
}
```

Code 33: JavaScript tp_drop Funktion

Anschließend, wird in Code 34 ein AJAX-Request an die Datei `tagesplan_add_employee_to_posten.php` geschickt, welche bestehende Einträge aktualisiert und neue vornimmt.

Das Element wird sofort zu dem Posten verschoben, der auf der Loslass-Fläche liegt. Außerdem werden die absoluten Positionen des Elements auf `0` gesetzt, weil sich auch die absolute Position zu einer Relativen geändert hat (jQuery-Skript übernimmt dies), und das Element sonst einfach irgendwo platziert wären würde.

```
function update_posten_position(e,ui){
    if($(this)[0]=== $(ui.item.parent())[0])
        $.ajax({
            type:"POST",
            url:"ajax/tagesplan_update_step_position.php",
            data:{
                'date': $('#tagesplan').data('date'),
                'stepid': $(ui.item).data('stepid'),
                'orderid': $(ui.item).data('id'),
                'group': $(ui.item).parent().index()+1
            }
        });
}
```

Code 34: JavaScript update_posten_positionen Funktion

11.4. Back-End

Mit Back-End wird im groben Sinne die Logik hinter einer Applikation bezeichnet, die dafür sorgt, dass die Applikation möglichst stabil läuft. Dazu verwaltet sie Daten, und ist Dreh- und Angelpunkt für alle möglichen Aktionen des Benutzers.

11.4.1. Template

Ein Template ist wichtig für die schnelle Darstellung von Inhalten. Man spart viel Zeit und versteckt alle wichtigen Funktionen in einer kleinen Datei, um diese danach nurnoch von außen aufzurufen. Das macht die einzelnen Dateien kleiner und übrsichtlicher.

```
private $pageIndex = null;
public $title = null, $permission = null, $pages = array();

public function __construct() {
    define('__CONTENT','cntnt');
    $this->setPages();
    $this->setTitle();
    if(!isset($_GET['p']) && !isset($_SESSION['id']))
        $this->pageIndex = 'login';
    else if(!isset($_GET['p']) && isset($_SESSION['id']))
        $this->pageIndex = 'wochenplan';
    else
        $this->pageIndex = trim($_GET['p']);
}
```

Code 35: Klasse Template mit Konstruktorfunktion

Die Klasse `SQL` ist tatsächlich in beinahe jedem PHP-Skript des Back-Ends und Front-Ends zu finden. Daher wird ab diesem Abschnitt nicht mehr näher darauf eingegangen.

11.4.1.1. Konstruktor

Wenn eine Instanz einer Klasse erstellt wird, so ist die erste Funktion die Aufgerufen wird stets `__construct()`. Das spart bei jeder neuen Instanzierung einen Funktionsaufruf

einer äquivalenten Initialisierungsfunktion, wie zB: `$instace->init()`, welche man vorher natürlich in der Klasse definieren muss[39].

In der Funktion `__construct()` wird die Konstante[40] `__CONTENT` mit dem Namen des Verzeichnisses definiert, in welchem sich alle Front-End Dokumente der Module befinden. Anschließend werden die Funktionen `setPages()` (siehe 11.4.1.2) und `setTitle()` (siehe 11.4.1.4) aufgerufen.

Nun wird der Wert für die von außen von Zugriffen geschützten Variable `pageIndex` ein Wert, abhängig von der momentanen Berechtigung des Benutzers und des Vorhandenseins der `$_GET`-Variable oder des Wertes am Index `p` der `$_GET`-Variable vergeben.

11.4.1.2. Funktion setPages()

Sie ist zuständig, die wichtigsten Informationen über eine Seite in einem Array zu organisieren. Die Datensätze werden von Hand eingegeben, weil das sicherer von unerlaubten Zugriffen ist.

```
private function setPages() {
    $this->pages = array(
        'wochenplan' => array(
            'title' => 'Wochenplan',
            'file' => '/wochenplan.php',
            'nav' => true,
            'permission' => 'wp'
        ),
        'dienstplan' => array(
            'title' => 'Dienstplan',
            'file' => '/dienstplan.php',
            'nav' => true,
            'permission' => 'dp'
        ),
        ...
    );
}
```

Code 36: Klasse Template mit setPages()-Funktion

Die Variable `pages` wird in Code 36 mit einem multidimensionalen Array, das an den Stellen `wochenplan`, `dienstplan`, etc. ein weiteres Array enthält definiert. Dieses Sub-Array enthält Eigenschaften einer Seite im Templatesystem.

Der Wert des Indexes `title` legt den Titel der Seite im HTML-Dokument Fest und erscheint in folgender (Werkseinstellung) Form: Planify • `[title]`.

[39] http://www.php.net/manual/en/language.oop5.decon.php (19.02.2014)
[40] http://si1.php.net/manual/en/function.define.php (05.05.2014)

Wichtig für die Einbindung der Datei in das HTML-Dokument mit `$template->loadPage()` ist der Wert des Arrays am Index `file`. Er Verweist auf ein Dokument im `__CONTENT`-Verzeichnis.

Ist der boolsche des Werts am Index `nav` gleich `true`, besteht die Möglichkeit einen Link auf dieses Dokument im Navigationsbereich des Dokuments vorzufinden. Diese Einstellung ist aber sehr stark von den Berechtigungen des momentan angemeldeten Benutzers abhängig.

Am Index `permission` findet sich, ganz entgegen aller Vermutungen, kein boolscher Wert oder ein Integer. Stattdessen ist der Wert vom Datentyp String und ist das Kürzel der Seite in der Berechtigungstabelle in der Datenbank.

11.4.1.3. Funktion hasPermission()

Diese Funktion überprüft mithilfe des Kürzels am Index permission des Arrays pages, ob der Benutzer das überhaupt im System angemeldet ist und ob dieser die Berechtigung hat die aufgerufene Seite überhaupt sehen zu dürfen.

Ist der Benutzer berechtigt die Seite aufzurufen, wird der boolsche Wert `true` zurückgegeben, wenn er keine Berechtigung hat, wird `false` zurückgegeben.

```
private function hasPermission($acronym = null) {
    $db = new SQL();
    if(isset($_SESSION['id']) && !is_null($acronym)) {
        $db->query('SELECT `ID` FROM `pf_login` WHERE `'. $acronym.'` IS
        NOT NULL AND `ID` = :id',array(
            ':id' => mysql_real_escape_string($_SESSION['id'])
        ));
        return ($db->count > 0 ? true : false);
    } else if(!isset($_SESSION['id'])) {
        return false;
    } else {
        return true;
    }
}
```

Code 37: Klasse Template mit hasPermission-Funktion

11.4.1.4. Funktion setTitle()

Die in Code 38 zu findende Funktion macht im Grunde genommen nichts anderes, als die den Wert an der Stelle `pageIndex` des Arrays `pages` abzufragen und nach dem zurückgegebenen Ergebnis zu handeln. Ist der Wert vorhanden, wird die Variable auf den im Array gefunden Wert festgelegt. Wird nichts gefunden, wird der Titel an der Stelle `err404`, also der Fehlerseite genommen. Wenn der `pageIndex` (überraschend) leer sein, wird der Titel der Standardseite (also Wochenplan) hergenommen. Ist er jedoch nicht gesetzt, wird abermals der Titel der Fehlerseite als Wert herangezogen.

```
private function setTitle() {
    if(isset($this->pageIndex)) {
        if(!empty($this->pages[$this->pageIndex]['title']))
            $this->title = $this->pages[$this->pageIndex]['title'];
        else
            $this->title = $this->pages['err404']['title'];
    } else if(empty($this->pageIndex))
        $this->title = $this->pages['wochenplan']['title'];
    else
        $this->title = $this->pages['err404']['title'];
}
```

Code 38: Klasse Template mit SetTitle-Funktion

11.4.1.5. Funktion loadPage()

Die Funktion in Code 38 überprüft, ob die Variable `pageIndex` gesetzt ist und ob an der Stelle des Wertes von `pageIndex` im Array `pages` überhaupt etwas zu finden ist und ob die dort angegebene Datei überhaupt im System exisitiert.

Ist das der Fall und hat der Benutzer die passende Berechtigung zu der Seite, wird sie an Ort und Stelle inkludiert, also übernommen, hat der Benutzer jedoch keine Berechtigung und die Seite exisitiert, wird die Datei an der Stelle `err500` im Array `pages` aufgerufen. Fehlt die Angabe über das Berechtigungskürzel für die Datenbank, wird davon ausgegangen, dass die Seite öffentlich zugänglich ist und wird inkludiert.

Wenn die Variable `pageIndex` nicht gesetzt ist, an der Stelle des Wertes von `pageIndex` im Array `pages` nichts zu finden ist oder das dort angegeben Dokument am Server nicht exisitert, wird standardmäßig das Fehlerdokument eingebunden, das im Normalfall immer vorhanden ist.

```
public function loadPage() {
    if(isset($this->pageIndex) && isset($this->pages[$this->pageIndex])
    && file_exists(__CONTENT . $this->pages[$this->pageIndex]['file'])) { //
VALID
        if(isset($this->pages[$this->pageIndex]['permission']))
            if($this->hasPermission($this->pages[$this-
>pageIndex]['permission']))
                // MySQL => HAT BERECHTIGUNG
                include_once( __CONTENT . $this->pages[$this->pageIndex]['file']
);
            else
                // MySQL => NICHT
                include_once( __CONTENT . $this->pages['err500']['file'] );
        else
            include_once( __CONTENT . $this->pages[$this->pageIndex]['file'] );
    } else {
        include_once( __CONTENT . $this->pages['err404']['file'] );
    }
}
```

Code 39: Klasse Template mit loadPage-Funktion

11.4.2. SQL-Klasse

Es ist das ungeschönte Herzstück der Web-Applikation und sorgt jedes Mal für eine wunderbar reibungslose Verbindung mit der MySQL-Datenbank. Hinter ihr stecken viele Stunden Arbeitszeit und noch mehr an Erfahrung und Technologiewissen. Sie trägt zu einer effektiven Query-Entwicklung bei und kommt auch in der fertigen, Production-Version zum Einsatz.

11.4.2.1. Konstruktor[41]

In Bezug auf Code 40 erkennt man innerhalb des Standardkonstruktors, dass versucht wird über den Data Source Name (DSN), also alle Datenbankspezifischen Informationen, den Usernamen und das zugehörige Passwort eine MySQL-Verbindung aufzubauen. Für dieses Projekt wurde die PDO-Schnittstelle verwendet.

Anschließend sagt man an der Schnittstelle dem Treiber noch mittels `PDO::ATTR_EMULATE_PREPARES`, dass man mehrere Queries auf einmal ausführen möchte, das ist wichtig, um so noch mehr Abfragen in nur eine einzige Verbindung zu packen.

Schlägt die Verbindung fehl, wird die Variable `e` mit einer Fehlermeldung beschrieben, welche in der Web-Applikation später ausgelesen werden kann.

11.4.2.2. Query-Funktion

Die Funktion benötigt 2 Parameter `$sql` (Auszuführender Query mit Variablen) und `$arrParams` (Variablen mit zugehörigen Werten in Arrayform). Anhand dieser wird dem SQL-Statement (`sth`) die Datenbankverbindung (`dbh`) mitsamt dem vorher präparierten Query zugewiesen, welche dann über `execute()` zusammen mit dem Parameterarray ausgeführt wird.

Das Ergebnis wird dann in der Variable `row` mithilfe der Funktion `fetchAll`[42] des Statements gespeichert. Das Ergebnis ist stark abhängig von dem Query und wird in Arrayform gespeichert. Die Funktion ist auch nicht geeignet um riesige Datenmengen abzufragen, in dem Benutzerhandbuch wird ausdrücklich darauf hingewiesen, dass man Ergebnisse vorher mit den MySQL-Anweisungen `WHERE` und `ORDER BY` sortieren soll, bevor man selbiges mit PHP macht.

Anschließend wird die Anzahl der gefundenen Datensätze des Statements in die Variable `count` gespeichert, die nach so ziemlich jeder Abfrage des Applikationscodes gebraucht wird. Danach wird der Variable `lastID` der Integer-Wert der letzten `ID` zugewiesen, was bei vielen Punkten des Projekts sehr hilfreich war und so ziemlich der Hauptgrund für die Wahl einer PDO-Schnittstelle anstatt einer herkömmlichen MySQLi-Schnittstelle, bei der

[41] http://www.php.net/manual/de/pdo.construct.php
[42] http://www.php.net/manual/de/pdostatement.fetchall.php

man zwar selbiges über die MySQL-Anweisung `LAST_INSERT_ID()` erreichen könnte, aber das behindert die Query-Entwicklung.[43]

```
class SQL {
    private $dbh = null, $sth = null;
    public $row = array(), $count = null, $error = null, $e = null,
$lastID = null, $success = false;
    public function __construct() {
        try {
            $this->dbh = new
PDO('mysql:dbname=planifydb;host=127.0.0.1;charset=utf8','user','pass');
            $this->dbh->setAttribute(PDO::ATTR_EMULATE_PREPARES, 1);
        } catch(PDOException $e) {
            $this->e = 'Verbindung abgebrochen: ' . $e->getMessage();
        }
    }
    public function query($sql, $arrParams) {
        try {
            $this->sth = $this->dbh->prepare($sql, array(PDO::ATTR_CURSOR
=> PDO::CURSOR_FWDONLY));
            $this->sth->execute($arrParams);
            $this->row = $this->sth->fetchAll();
            $this->count = $this->sth->rowCount();
            $this->error = $this->sth->errorInfo();
            $this->lastID = $this->dbh->lastInsertId();
        } catch(PDOExecption $e) {
            $this->dbh->rollback();
            $this->e .= 'Error: ' . $e->getMessage();
        }
    }
}
```

Code 40: SQL-Klasse

11.4.3. Komponenten

11.4.3.1. Wochenplan

In den folgenden Punkten werden Komponenten des Wochenplans (siehe Abb. 37), unterteilt in Eingangslogik (siehe Code 41), MySQL-Statements und Ausgabe, erklärt.

```
if(isset($_GET['s']) && preg_match('/(\d{4})(\d{2})(\d{2})/i',
$_GET['s'], $matches))
    $pdate = mktime(0,0,0,$matches[2],$matches[3],$matches[1]);
else
    $pdate = time();
$header = new Header();
$main = new Main();
```

Code 41: Eingangslogik des Wochenplans

Der Sourcecode in Code 42 benötigt eine Zeitangabe, um den vom Benutzer angeforderten Zeitraum zu liefern. Die Kalenderwoche wird über die `$_GET`-Variable am

[43] http://www.php.net/manual/de/class.pdo.php (25.02.2014)

Index `s` im Format YYYYMMDD gesetzt. Kalenderwoche 11 wird in Form von`?s=20140310` angegeben.Der Reguläre Ausdruck in Zeile 3 sucht nach genau diesem Muster. Wird dieses erkannt, wird ein sog. Timestamp dieses Datums erzeugt, also der Zeitstempel 2014-03-10 um 00:00:00. Wird das Muster nicht erkannt, wird einfach der aktuelle Zeitstempel (gerade in diesem Moment) genommen.

```
$header->arrLinks[] = 'index.php?p=wochenplan&s='.
    date("Ymd",strtotime(date("d.m.Y",$pdate).'-1 week'));
$header->arrLinks[] = 'index.php?p=wochenplan&s='.
    date("Ymd",strtotime(date("d.m.Y",$pdate).'+1 week'));
```

Code 42: Berechnung des Datums von vorheriger/nächster Woche

Der Code aus Code 42 berechnet einen Tag aus der vorherigen/nächsten Woche mittels der vorher in Code 41 definierten Variable. Die Funktion `date()` nimmt 2 Argumente an. Das erste legt fest, in welcher Form das Datum ausgegeben werden soll. Es wurde die Form `YYYYMMDD`, also das Parameter `"Ymd"` gewählt.

Da ein Timestamp nichts anderes als die vergangenen Sekunden seit dem 1.1.1970 00:00:00 sind, kann man auch ganz einfach `$pdate-3600*24*7` rechnen, also 3600 Sekunden einer Stunde mal 24 Stunden am Tag mal 7 Tage die man zurückgehen möchte.

```
SELECT
        NULL as "ID",
        NULL as "stepID",
        Q.`dayIndex` as "i",
        NULL as "quantity",
        P.`name`, "ffffff" as "color",
        0 as "duration",
        "Lieferant" as "productName"
    FROM
        `pf_lieferanten` P
        JOIN `pf_lieferanten_zuteilungen` Q
    WHERE
        Q.`supplierID` = P.`ID` AND
        CEIL((WEEKOFYEAR(:date)-WEEKOFYEAR(P.`date`))%(P.`period`+1)) = 0
UNION
SELECT
        NULL as "ID",
        NULL as "stepID",
        WEEKDAY(DATE_ADD(P.`birthday`,INTERVAL (YEAR(:date)-
YEAR(P.`birthday`))  YEAR)) as "i",
        (YEAR(CURDATE())-YEAR(`birthday`)+1) as "quantity",
        P.`firstname`,
        "ffffff" as "color",
        0 as "duration",
        "Geburtstag" as "productName"
    FROM
        `pf_mitarbeiter` P
    WHERE
        DAYOFYEAR(`birthday`) >= DAYOFYEAR(FIRST_DAY_OF_WEEK(:date)) AND
        DAYOFYEAR(`birthday`) <= DAYOFYEAR(FIRST_DAY_OF_WEEK(:date))+6
UNION
```

Code 43: SQL-Statementder Produkte des Wochenplans Teil 1

```
SELECT
        P.`ID`,
        Q.`ID` as "stepID",
        WEEKDAY(DATE_ADD(P.`date`,INTERVAL (Q.`dayIndex`-1) DAY)) as "i",
        T.`quantity`,
        Q.`name`,
        U.`color`,
        TIME_TO_SEC(S.`time`) as "duration",
        R.`name` as "productName"
    FROM
        `pf_bestellungen` P JOIN `pf_produkte_arbeitsschritte` Q
        JOIN `pf_produkte` R JOIN `pf_produkte_zeiten` S
        JOIN `pf_produkte_mengen` T JOIN `pf_colors` U
    WHERE
        U.`ID` = R.`colorID` AND
        T.`ID` = P.`qtyID` AND
        S.`stepID` = Q.`ID` AND
        S.`qtyID` = P.`qtyID` AND
        R.`ID` = P.`productID` AND
        Q.`productID` = P.`productID` AND
        DATE_ADD(P.`date`,INTERVAL (Q.`dayIndex`-1) DAY) >=
FIRST_DAY_OF_WEEK(:date) AND
        DATE_ADD(P.`date`,INTERVAL (Q.`dayIndex`-1) DAY) <=
DATE_ADD(FIRST_DAY_OF_WEEK(:date), INTERVAL 6 DAY) AND
        NOT EXISTS(SELECT * FROM `pf_wochenplan_bestellungen` WHERE
`stepID` = Q.`ID` AND `orderID` = P.`ID`)
UNION
SELECT
        S.`ID`,
        P.`stepID`,
        WEEKDAY(P.`date`) as "i",
        (SELECT `quantity` FROM `pf_produkte_mengen` WHERE `ID` =
S.`qtyID`) as "quantity",
        Q.`name`,
        V.`color`,
        TIME_TO_SEC(U.`time`),
        R.`name` as "productName"
    FROM
        `pf_wochenplan_bestellungen` P JOIN `pf_produkte_arbeitsschritte`
Q JOIN `pf produkte` R
        JOIN `pf_bestellungen` S JOIN `pf_produkte_mengen` T JOIN
`pf_produkte_zeiten` U JOIN `pf_colors` V
    WHERE
        V.`ID` = R.`colorID` AND
        U.`productID` = S.`productID` AND
        U.`stepID` = Q.`ID` AND
        U.`qtyID` = S.`qtyID` AND
        T.`ID` = S.`qtyID` AND
        S.`ID` = P.`orderID` AND
        R.`ID` = Q.`productID` AND
        Q.`ID` = P.`stepID` AND
        S.`date` IS NOT NULL AND
        P.`date` <= DATE_ADD(FIRST_DAY_OF_WEEK(:date), INTERVAL 6 DAY)
        AND P.`date` >= FIRST_DAY_OF_WEEK(:date)
```

Code 44: SQL-Statementder Produkte des Wochenplans Teil 2

Eine Beispielabfrage, mit dem SQL-Query aus Code 43 & Code 44 ist in Abb. 40 abgebildet. Dabei wird die Variable `:date` durch den Wert von `$pdate` ersetzt, dessen Wert vorher festgelegt wird.

ID	stepID	i	quantity	name	color	duration	productName
NULL	NULL	1	NULL	Alpha	ffffff	0	Lieferant
NULL	NULL	4	NULL	Breganzia	ffffff	0	Lieferant
113	31	0	400	Aluformen mit Mürbteig auslegen + backen	06A2CB	24000	Candy Bliss Schokotarte
113	32	1	400	Schoko-creme abfüllen	06A2CB	7200	Candy Bliss Schokotarte
113	33	2	400	Abglänzen und verpacken	06A2CB	10800	Candy Bliss Schokotarte

Abb. 40: Beispielabfrage aus Code 43.

Die Spalte `i` repräsentiert den Tag in der Woche, also den index des Tages an dem dieser Schritt stattfindet. Dieser ist natürlich abhängig ob der Datensatz ein Lieferant, Geburtstag oder ein Schritt in dem Produktzyklus ist. Man geht folgendermaßen vor:

Zuerst werden in Code 43 alle Lieferanten mit zugehörigen Einteilungen herausgesucht, anschließend werden die Geburtstage aller Mitarbeiter angefügt. In Code 44 werden dann reguläre Bestellungen herausgesucht unter berücksichtigung, dass diese nicht schon in der Tabelle `pf_wochenplan_bestellungen` zu finden sind. Dann werden nurnoch diese aus der Tabelle `pf_wochenplan_bestellungen` angefügt und man hat eine Abfrage, die somit Lieferanten, Geburtstage, reguläre Bestellungen und verschobene Bestellungen abdeckt.

Das in Code 46 beschriebene SQL-Statement sucht in der Datenbank nach allen Mitarbeitern und Leiharbeitern, die an allen Tagen der ausgewählten Woche Dienst haben, um so die verfügbare Mitarbeiterzeit an diesem Tag zu berechnen.

Dafür wird zuerst der Tagesindes berechnet. Mittels `DATE_DIFF(date arg1, date arg2)` kann man die Differenz in (int) Tagen berechnen. Dazu wird das Datum des ersten Tages der Woche `FIRST_DAY_OF_WEEK(date Datum)`, von dem Datum der Zuteilung `P.`date`` subtrahiert.

Die Funktion `FIRST_DAY_OF_WEEK` gibt im Grunde genommen nur die Berechnung der `DATE_SUB(date arg1, int arg2)`-Funktion aus. Diese Nimmt das mitgelieferte Datum `arg1` und geht um `arg2`-Schritte zurück, in diesem Fall um genau so viele, wie der Tagesindex des mitgelieferten Tages ist, also der `WEEKDAY(date arg1)`-Index.

```
DELIMITER $$
CREATE FUNCTION `FIRST_DAY_OF_WEEK`(day DATE) RETURNS date
    DETERMINISTIC
BEGIN
  RETURN SUBDATE(day, WEEKDAY(day));
END$$
DELIMITER;
```

Code 45: Funktion FIRST_DAY_OF_WEEK in MySQL

Insbesondere die Anweisung `DELIMITER` ist neu gewesen, deren Funktion war jedoch schnell gelernt und angewandt. Mit `DELIMITER` verändert man das Begrenzungszeichen Clientseitig von (Standard) `;` in `$$`, um die ganze Funktion in ein Statement zu bekommen.[44]

```
SELECT
        DATEDIFF(P.`date`,FIRST_DAY_OF_WEEK(:date)) as "i",
        P.`ID`,
        (Q.`grop`-1) as "pID",
        FLOOR((TIME_TO_SEC(IF(P.`endtime` IS NULL, Q.`endtime`,
P.`endtime`))-TIME_TO_SEC(IF(P.`starttime` IS NULL, Q.`starttime`,
P.`starttime`)))/60) as "total"
    FROM
        `pf tagesplan zuteilungen` P JOIN `pf mitarbeiter` Q
        JOIN `pf_berufsgruppen` R JOIN `pf_colors` S
    WHERE
        S.`ID` = R.`colorID` AND
        R.`ID` = Q.`group` AND
        Q.`ID` = P.`employeeID` AND
        P.`temp` IS NULL AND
        P.`date` >= FIRST_DAY_OF_WEEK(:date) AND
        P.`date` <= DATE_ADD(FIRST_DAY_OF_WEEK(:date), INTERVAL 6 DAY)
UNION
SELECT
        DATEDIFF(P.`date`,FIRST_DAY_OF_WEEK(:date)) as "i",
        P.`ID`,
        3,
        FLOOR((TIME_TO_SEC(P.`endtime`)-TIME_TO_SEC(P.`starttime`))/60)
as "total"
    FROM
        `pf_tagesplan_zuteilungen` P JOIN `pf_leiharbeiter` Q
        JOIN `pf_berufsgruppen` R JOIN `pf_colors` S
    WHERE
        S.`ID` = R.`colorID` AND
        P.`temp` IS NOT NULL AND
        Q.`ID` = P.`employeeID` AND
        R.`ID` = 4
    ORDER BY
        `i`,
        `pID`
    ASC
```

Code 46: Mitarbeiterzeiten-Query im Wochenplan

Der letzte verwendete Query im Wochenplan sucht nach den Bestellten Produkten mit Zeiten und Mengen und ist in Code 46 festgehalten. Dabei zu beachten ist, dass die Bestellungen unabhängig von der ausgewählten Woche angezeigt werden.

[44] http://dev.mysql.com/doc/refman/5.1/de/create-procedure.html (05.03.2014)

```
SELECT
    P.`ID`,
    R.`quantity`,
    S.`color`,
    Q.`name`
FROM
    `pf_bestellungen` P JOIN `pf_produkte` Q
    JOIN `pf_produkte_mengen` R JOIN `pf_colors` S
WHERE
    S.`ID` = Q.`colorID` AND
    R.`ID` = P.`qtyID` AND
    P.`date` IS NULL AND
    Q.`ID` = P.`productID`
ORDER BY
    Q.`name`
ASC
```

Code 47: SQL-Abfrage der Produkte im Wochenplan

Die Zeilen 114 f. werden nun vereinfacht dargestellt, weil es nicht des ganzen Codes bedarfs. Für detailliertere Informationen werfen Sie bitte einen Blick auf den beigefügten Datenträger und betrachten Zeilen 114 f. in `source/cntnt/wochenplan.php`.

Code 49 zeigt (vereinfacht), wie die beweglichen/statischen Elemente des Wochenplans erzeugt werden. Es werden 7 Spalten mittels `for()`-Schleife erzeugt und erst in dieser nachgeschaut, ob Elemente angezeigt werden sollen. Der Trick dabei ist es, die Daten aus der Datenbank so aufzubereiten, dass man mit nur einer einzigen Abfrage den gesamten Wochenplan erhält ohne 7 Mal zu prüfen, ob etwas vorhanden ist.

Das Standard `$db->row`-Array ist dabei nicht hilfreich. Es wird stattdessen ein Array`$data`erstellt und durch die `for`-Schleife befüllt. Undzwar wird an der Stelle `$data[x]` nun ein neues Array erzeugt, wobei `x` dem Tagesindex `i` aus der Datenbank entspricht.

```
$data = array();
for($db->row as $i => $row)
    $data[$row['i']][] = $row;
```

Code 48: Datenaufbereitung für den Wochenplan

Da die Daten nun wunderbar aufbereitet sind, muss man diese nurnoch auf Länge überprüfen, ist diese größer 0, wird damit begonnen alle Schritte dieses Arrays auszugeben. Die Eigenschaften `color`, `ID`, `stepID`, `duration`, `qty` und `productName` sind wichtig für den Schritt, und ersetzen (natürlich) die Fragezeichen. Außerdem werden noch der Schrittname, Menge und Produktname dem Benutzer sichtbar angezeigt.

```
<div class="tr">
<!-- 7 SPALTEN -->
<ul class="unstyled employeelist hide sortable" data-date="YYYY-MM-DD">
    <li class="text-center day">
        <a href="index.php?p=tagesplan&s=YYYYMMDD">
            <span>0</span>
            <div>MA:
                <span><!-- MA-Zeit --></span>
                <span class="additi"><!-- LEIHARBEITER? --></span>
            </div>
            <div>
                <strong><!-- TAGNAME --> <!-- DD.MM. --></strong>
            </div>
        </a>
    </li>
    <!-- AUSGABE ALLER ARBEITSSCHRITTE, GEB.-TAGE usw. -->
    <li class="item box" style="background-color:#??????" data-id="??"
data-stepid="??" data-duration="??" data-qty="??" data-prodname="????">
        <div class="overflow"><!-- SCHRITTNAME --></div>
        <span class="fancy qty"><!-- MENGE --> <!-- PRODUKTNAME --
></span>
    </li>
</ul>
<!-- ENDE 7 SPALTEN -->
<ul class="unstyled hide orders">
    <li class="text-center">
        <div> </div>
        <strong>
            <div>Bestellungen</div>
        </strong>
        <span><!-- ANZAHL BESTELLUNGEN --></span>
    </li>
    <!-- BESTELLUNGEN -->
    <li class="eitem" data-id="??" style="background-color:#??????">
        <div class="overflow"><!-- PRODUKTNAME --></div>
        <span class="fancy qty"><!-- MENGE --></span>
    </li>
    <!-- ENDE BESTELLUNGEN -->
</ul>
```

Code 49: Markup Wochenplan Arbeitsschrite + Bestellungen (vereinfacht)

11.4.3.2. Dienstplan

Der Aufbau des Dienstplans (siehe 9.2) ähnelt stark dem des Wochenplans, so sind der Header (bis auf den Text) und die Toolbar (rechts unten) von der Programmierung ident. Zur Vermeidung von Redundanzen wird dieser Teil nicht genauer erläutert.

```php
$employees = new SQL();
$employees->query('SELECT `ID` FROM `pf_mitarbeiter`',array());
// VORERST NUR AKTUELLE WOCHE
$db->query('SELECT `ID` FROM `pf_tagesplan_zuteilungen` WHERE `date` >=
FIRST_DAY_OF_WEEK(:date) AND `date` <= DATE_ADD( FIRST_DAY_OF_WEEK(:date),
INTERVAL 6 DAY)',array(
    ':date' => mysql_real_escape_string(date("Y-m-d",$pdate))
));
$query = '';
if($db->count == 0) { // NO EMPLOYEES FOUND, NO PROBLEM => CALL THEM TO WORK!
    for($i=0;$i<6;$i++) {
        $date = strtotime($time->row[0]["date"].' +'.$i.' day');
        if(!$main->isHoliday(date("d.m.Y",$date)))
            foreach($employees->row as $j => $employee)
                $query .= 'INSERT IGNORE INTO `pf_tagesplan_zuteilungen`
(`date`,`employeeID`) VALUES("'.date("Y-m-d",$date).'",'.$employee['ID'].'); ';
    }
}
// EINTEILEN
$db->query($query,array());
```

Code 50: Mitarbeiter einteilen im Dienstplan

Wie in Code 50 zu sehen ist, wird zuerst abgefragt, ob schon Mitarbeiter der aktuellen Kalenderwoche zugewiesen sind. Sind keine zugeteilt, wird für Montag-Samstag (=6 Tage) das zugehörige Datum errechnet und ein Query zusammengesetzt, der jeden Mitarbeiter an jeden dieser Tage einteilen soll, vorausgesetzt der aktuelle Tag ist kein Feiertag [`$main->isHoliday()`]. Später wird dieser zusammengesetzte Query noch ausgeführt und die Mitarbeiter sind nun eingeteilt.

```
SELECT
        DATEDIFF(P.`date`,FIRST_DAY_OF_WEEK(:date)) as "i",
        P.`ID`,
        P.`date`,
        P.`employeeID` as "eID",
        IF(P.`boss` IS NOT NULL,1,NULL) as "boss",
        Q.`firstname`,
        Q.`surname`,
        Q.`image`,
        (Q.`group`-1) as "pID",
        IF(P.`starttime` IS NULL, Q.`starttime`, P.`starttime`) as
"starttime",
        IF(P.`endtime` IS NULL, Q.`endtime`, P.`endtime`) as "endtime",
        S.`color`
    FROM
        `pf_tagesplan_zuteilungen` P JOIN `pf_mitarbeiter` Q
        JOIN `pf_berufsgruppen` R JOIN `pf_colors` S
    WHERE
        S.`ID` = R.`colorID` AND
        R.`ID` = Q.`group` AND
        Q.`ID` = P.`employeeID` AND
        P.`temp` IS NULL AND
        P.`date` >= FIRST_DAY_OF_WEEK(:date) AND
        P.`date` <= DATE_ADD(FIRST_DAY_OF_WEEK(:date), INTERVAL 6 DAY)
UNION
SELECT
        DATEDIFF(P.`date`,FIRST_DAY_OF_WEEK(:date)) as "i",
        P.`ID`,
        P.`date`,
        P.`employeeID`,
        P.`boss`,
        Q.`name`,
        NULL,
        NULL,
        3,
        P.`starttime`,
        P.`endtime`,
        S.`color`
    FROM
        `pf_tagesplan_zuteilungen` P JOIN `pf_leiharbeiter` Q
        JOIN `pf_berufsgruppen` R JOIN `pf_colors` S
    WHERE
        S.`ID` = R.`colorID` AND
        P.`temp` IS NOT NULL AND
        Q.`ID` = P.`employeeID` AND
        R.`ID` = 4
    ORDER BY
        `i`,
        `pID`
    ASC
```

Code 51: SQL-Abfrage Dienstplan Mitarbeiter

Der Index des Mitarbeiters in Code 51 wird wie im Wochenplan schon erklärt berechnet. Die Ergebnisse des ersten `SELECT`-Statements werden mit denen einen zweiten vereint.

Dieses fragt jedoch nur Leiharbeiter ab, anschließend werden beide Ergebnisse nach dem Tagesindex `i` und `pID` in aufsteigender Reihenfolge sortiert.

```
$days = array("Mo., ", "Di., ", "Mi., ", "Do., ", "Fr., ", "Sa., ", "So., ");
// SQL-ABFRAGE DIENSTPLAN MITARBEITER
if($db->count > 0)
    foreach($db->row as $row)
        $arr[$row['i']][] = $row;
for($i=0;$i<7;$i++) {
    $date = strtotime($time->row[0]['date'].' +'.$i.' day');
    echo '<ul class="unstyled droppable';
    echo ($main->isHoliday(date('d.m.Y',$date))?' holiday':'');
    echo '" data-date="'.date("Y-m-d",$date).'">
        <li class="text-center day">
            <a href="index.php?p=tagesplan&s='.date("Ymd",$date).'" class="pull-left">
                <i class="icon-eye-open"></i>
            </a>
            '.$days[$i].date("d.m.",$date).'
            <i class="icon-chevron-down animation pull-right"></i>
        </li>';
    if(isset($arr[$i]) && is_array($arr[$i]))
        foreach($arr[$i] as $item)
            echo '<li class="item draggable" data-pid="'.$item['pID'].'" data-eid="'.$item['eID'].'" data-id="'.$item['ID'].'" data-starttime="'.$item['starttime'].'" data-endtime="'.$item['endtime'].'" style="border-right:4px solid#'.$item['color'].'">
                <img src="img/u/'.$item['image'].'.JPEG" class="img-circle" />'.$item['firstname'].' '.$item['surname'].'
            </li>';
    echo '</ul>';
}
```

Code 52: PHP-Code zum Anzeigen des Dienstplans

Auch hier, bei Code 52 zu sehen, wird dasselbe Anzeigeprinzip, wie bei Code 49 angewandt. Man gibt 7 Mal eine Liste aus, egal ob sie gefüllt wird oder nicht. Abhängig von ihrem Datum fällt sie auf einen Feiertag oder nicht.

Später wird überprüft, ob sich ein Array an der Stelle `$i` befindet, anschließend wird mit den enthaltenen Daten einer jeden Stelle ein Mitarbeiterblock in das Dokument ausgegeben. Ein Leiharbeiter hat standardmäßig kein Bild, daher wird das Bild `img/u/u_unknown.JPEG` eingebunden. Doch nur theoretisch, denn die .htaccess Datei verhindert solche Peinlichkeiten. Wie in Code 53 zu sehen ist, werden alle Aufrufe, die in das Verzeichnis `img/u/` gehen abgefangen und an die Datei `imagerotate.php` weitergeleitet.

```
RewriteRule ^img/u/([^/]*)$ var/imagerotate.php?img=$1 [L]
```

Code 53: Auszug der .htaccess Datei

Der Dateiname wird in das Array `$_GET` unter dem Index `img` gespeichert.

```php
header('Content-type: image/jpeg');
function image_fix_orientation(&$image, $filename) {
    $image = imagerotate($image, array_values([0, 0, 0, 180, 0, 0, -90,
0, 90])
    [@exif_read_data($filename)['Orientation'] ?: 0], 0);
    return $image;
}
$filename =
(file_exists('../img/u/'.$_GET['img'])?'../img/u/'.$_GET['img']:'../img/u
/u_unknown.JPEG');
$source = imagecreatefromjpeg($filename);
$source_img = image_fix_orientation($source, $filename);
$new_w = $new_h = 100;

$orig_w = imagesx($source_img);
$orig_h = imagesy($source_img);

$w_ratio = ($new_w / $orig_w);
$h_ratio = ($new_h / $orig_h);

if ($orig_w > $orig_h ) {//landscape
    $crop_w = round($orig_w * $h_ratio);
    $crop_h = $new_h;
} elseif ($orig_w < $orig_h ) {//portrait
    $crop_h = round($orig_h * $w_ratio);
    $crop_w = $new_w;
} else {//square
    $crop_w = $new_w;
    $crop_h = $new_h;
}
$dest_img = imagecreatetruecolor($new_w,$new_h);
imagecopyresampled($dest_img,$source_img,0,0,0,0,$crop_w,$crop_h,$orig_w,
$orig_h);
imagejpeg($dest_img);
```

Code 54: Skript zum Manipulieren von Bildern mit PHP

Das Skript startet gleich damit, die Headerinformationen der PHP-Datei zu modifizieren. Ausgabetyp des Dokuments ist nun ein Bild im Format JPEG. Nach dem definieren des Dateinamens `$filename` (abhängig ob die Datei am Server gefunden wird, wichtig für Dienstplan!) wird eine temporäre Kopie des an der Stelle `$filename` am Server liegenden Bildes gemacht und die selbstdefinierte Funktion `image_fix_rotation()` darauf angewandt.

Das große Problem beim Aufnehmen von Bildern durch HTML5 ist nämlich, dass mobile Geräte gedreht werden können und damit auch die Fotos verdreht sind. Jedoch speichert jedes Mobilgerät in den sog. EXIF-Daten eines Bildes u.a. auch die Orientierung eben dieses. Diese muss man nurnoch abfragen und das Bild dementsprechend um 90° oder mehr drehen.

Danach werden die Breite des (gedrehten) Bildes mit `imagesx()` und die Höhe des (gedrehten) Bildes mit `imagesy()` ausgelesen. Das Bild soll nun nicht einfach verkleinert werden (wie es der HTML-Tag img machen würde). Es soll ein neues Bild erzeugt

werden, welches 100x100 Pixel groß ist, ohne dabei zu verzerren. Nach dem Ausrechnen der richtigen Faktoren wird ein neues Bild mit schwarzem Hintergrund[45] in den vorher festgelegten Dimensionen erzeugt.

Dieses wird anschließend mit dem beschnittenen, verkleinerten und gedrehten Bild überschrieben[46]. Nun wird ein JPEG Bild aus dieser Kopie erzeugt und im Browser ausgegeben[47].

11.4.3.3. Tagesplan

Der in Abb. 39 dargestellte Tagesplan wird zu einem Großteil durch den Query, der in Code 55 zu finden ist generiert. Dabei sind vor allem wieder der Index `i` des zugeteilten Posten, ob der zugeteilte Mitarbeiter eine verantwortliche Rolle ausübt (`boss`) und die Start- als auch Endzeit des Mitarbeiters. Die Abfrage wird intelligent verarbeitet. Ist der eingeteilte Arbeiter ein Mitarbeiter, so ist das Attribut temp der Tabelle `pf_tagesplan_zuteilungen` `NULL`, ist er jedoch nur ein Leiharbeiter, erhält `temp` den Wert `1`.

Es werden also oben nur zugeteilte Mitarbeiter gesucht an denen dann die zugeteilten Leiharbeiter angehängt werden.

[45] http://www.php.net/manual/de/function.imagecreatetruecolor.php (05.03.2014)
[46] http://www.php.net/manual/de/function.imagecopyresampled.php (05.03.2014)
[47] http://hu1.php.net/manual/en/function.imagejpeg.php (05.03.2014)

```
SELECT
        P.`ID`,
        P.`groupIndex` as "i",
        P.`date`,
        S.`color`,
        Q.`image`,
        IF(ISNULL(P.`boss`),"," boss") as "boss",
        Q.`firstname`,
        R.`ID` as "pID",
        Q.`ID` as "eID",
        Q.`surname`,
        IF(ISNULL(P.`starttime`),(SELECT `starttime` FROM
`pf_mitarbeiter` WHERE `ID` = Q.`ID`),P.`starttime`) as "starttime",
        IF(ISNULL(P.`endtime`),(SELECT `endtime` FROM `pf_mitarbeiter`
WHERE `ID` = Q.`ID`),P.`endtime`) as "endtime"
    FROM
        `pf_tagesplan_zuteilungen` P JOIN `pf_mitarbeiter` Q
        JOIN `pf_berufsgruppen` R JOIN `pf_colors` S
    WHERE
        S.`ID` = R.`colorID` AND
        Q.`ID` = P.`employeeID` AND
        R.`ID` = Q.`group` AND
        P.`temp` IS NULL AND
        P.`date` = :date
UNION
SELECT P.`ID`,
        P.`groupIndex` as "i",
        P.`date`,
        S.`color`,
        "u_unknown" as "image",
        NULL as "boss",
        Q.`name` as "firstname",
        T.`ID` as "pID",
        Q.`ID` as "eID",
        NULL as "surname",
        P.`starttime`,
        P.`endtime`
    FROM
        `pf_tagesplan_zuteilungen` P JOIN `pf_leiharbeiter` Q
        JOIN `pf_berufsgruppen` R JOIN `pf_colors` S
        JOIN `pf_berufsgruppen` T
    WHERE
        T.`istemp` IS NOT NULL AND
        S.`ID` = R.`colorID` AND
        R.`istemp` IS NOT NULL AND
        P.`temp` IS NOT NULL AND
        Q.`ID` = P.`employeeID` AND
        P.`date` = :date
```

Code 55: SQL-Abfrage Tagesplan Zuteilungen

Der Ablauf in Code 56 ist folgendermaßen: Ist der ausgewählte Tag kein Feiertag, werden die in Code 55 dargestellte SQL-Abfrage, eine Abfrage aller Posten und aller Berufsgruppen getätigt. Nun werden wie in 11.4.3.1 schon beschrieben, die zugeteilten Mitarbeiter nach dem Index ihres zugeteilten Postens sortiert.

```
if(!$main->isHoliday(date("d.m.Y",$pdate))) {
    // SQL-ABFRAGE TAGESPLAN ZUTEILUNGEN
    $db2->query('SELECT * FROM `pf_posten` ORDER BY `orderIndex`
ASC',array());
    $db3->query('SELECT Q.`color` FROM `pf_berufsgruppen` P JOIN
`pf_colors` Q WHERE Q.`ID` = P.`colorID` ORDER BY P.`ID` ASC',array());
    if($db->count > 0) // ZUTEILUNGEN GEFUNDEN
        foreach($db->row as $row)
            $posten[$row['i']][] = $row;
}
```

Code 56: Eingangslogik Tagesplan

Das in Code 55 beschriebene Verfahren zeigt die Suche nach Leiharbeitern in der Datenbank und ist besonders raffiniert. Man kann nämlich anhand der Lieferfrequenz (0=wöchentlich, 1=alle 2 Wochen, etc.) und einem Startdatum ausrechnen, wann der Lieferant kommt.

Das wird mithilfe von `(WEEK(:date)-WEEK(P.`date`))%(P.`period`+1)=0` geprüft. In der nächsten Abfrage wird auf existieren von demselben Schritt in pf_wochenplan_bestellungen geprüft. Ist dieser dort vorhanden wird für diese Teilabfrage vorerst nichts ausgegeben.

Anschließend werden in Code 57 die verschobenen Bestellungen ausgelesen (zerissene Produktketten usw.)

```
SELECT
        1 as "lieferant",
        DATE_ADD(:date, INTERVAL
DATEDIFF(DATE_ADD(FIRST_DAY_OF_WEEK(:date), INTERVAL Q.`dayIndex`
DAY),:date) DAY) as "date",
        NULL as "ID",
        NULL as "stepID",
        (SELECT `orderIndex` FROM `pf_posten` WHERE `isboss` IS NOT NULL)
as "group",
        "Lieferant" as "productName",
        P.`name`,
        0 as "quantity",
        0 as "duration",
        "eeeeee" as "color"
    FROM
        `pf_lieferanten` P JOIN `pf_lieferanten_zuteilungen` Q
    WHERE
        Q.`supplierID` = P.`ID` AND
        (WEEK(:date)-WEEK(P.`date`))%(P.`period`+1) = 0 AND
        DATE_ADD(:date, INTERVAL
DATEDIFF(DATE_ADD(FIRST_DAY_OF_WEEK(:date), INTERVAL Q.`dayIndex`
DAY),:date) DAY) >= :date
UNION
SELECT
        NULL as "lieferant",
        NULL as "date",
        P.`ID`,
        Q.`ID` as "stepID",
        V.`orderIndex` as "group",
        R.`name` as "productName",
        Q.`name`,
        S.`quantity`,
        TIME_TO_SEC(T.`time`) as "duration",
        U.`color`
    FROM
        `pf_bestellungen` P JOIN `pf_produkte_arbeitsschritte` Q
        JOIN `pf_produkte` R JOIN `pf_produkte_mengen` S
        JOIN `pf_produkte_zeiten` T JOIN `pf_colors` U JOIN `pf_posten` V
    WHERE
        V.`ID` = Q.`group` AND
        U.`ID` = R.`colorID` AND
        Q.`dayIndex` = (DATEDIFF(:date, P.`date`)+1) AND
        Q.`productID` = P.`productID` AND
        R.`ID` = P.`productID` AND
        T.`stepID` = Q.`ID` AND
        T.`qtyID` = P.`qtyID` AND
        S.`ID` = P.`qtyID` AND
        NOT EXISTS(SELECT * FROM `pf_wochenplan_bestellungen` WHERE
`stepID` = Q.`ID` AND `orderID` = P.`ID`)
UNION
```

Code 57: SQL-Abfrage Tagesplan Produktschritte Teil 1

```
SELECT
        NULL as "lieferant",
        NULL as"date",
        S.`ID`,
        P.`stepID` as "stepID",
        IF(ISNULL(P.`group`),X.`orderIndex`,P.`group`-1) as "group",
        R.`name` as "productName",
        Q.`name`,
        (SELECT `quantity` FROM `pf_produkte_mengen` WHERE `ID` =
S.`qtyID`) as "quantity",
        TIME_TO_SEC(V.`time`) as "duration",
        W.`color`
    FROM
        `pf_wochenplan_bestellungen` P JOIN `pf_produkte_arbeitsschritte`
Q
        JOIN `pf_produkte` R JOIN `pf_bestellungen` S
        JOIN `pf_produkte_mengen` T JOIN `pf_produkte_arbeitsschritte` U
        JOIN `pf_produkte_zeiten` V JOIN `pf_colors` W JOIN `pf_posten` X
    WHERE
        X.`ID` = Q.`group` AND
        W.`ID` = R.`colorID` AND
        V.`productID` = S.`productID` AND
        V.`stepID` = Q.`ID` AND
        V.`qtyID` = S.`qtyID` AND
        U.`ID` = P.`stepID` AND
        T.`ID` = S.`qtyID` AND
        S.`ID` = P.`orderID` AND
        R.`ID` = Q.`productID` AND
        Q.`ID` = P.`stepID` AND
        P.`date` = :date
```

Code 58: SQL-Abfrage Tagesplan Produktschritte Teil 2

Anhang

I. Pflichtenheft

Allgemeines

Allgemeine Informationen

Name der Mitglieder (alphabet.)	*Martin Thoma, Martin Weise*
Adresse	*2115 Simonsfeld, 2136 Laa / Thaya*
Telefon	*+4366475074706, +436603499411*
E-Mail	*Martin.Weise@gmx.at,thomamartin@gmx.at*

Tabelle 4: Pflichtenheft Kontaktinformationen

Allgemeine Projektdaten

Projektname	Planify App
Kunde	Heiss & Süß GmbH
Projektbetreuer	Paul Tutschek
Datum	09.04.2013
Version	1.0 - am 09.04.2013 2.0 - am 23.04.2013 3.0 - am 30.04.2013 4.0 - am 16.05.2013

Tabelle 5: Pflichtenheft Projektdaten

Einleitung

Das Pflichtenheft soll den Wünschen und Anforderungen der Firma Heiss & Süß GmbH entsprechen.

Das ist eine finale Version und ist später nicht mehr änderbar. Nachträgliche Änderungen werden als Vorschlag betrachtet und sind nicht verpflichtend.

Charakteristika

Ist-Zustand	In Planung
Soll-Zustand	Fertigstellung, Auslieferung
Allgemeine Ziele	Erstellen einer Webanwendung
Technische Ziele	Verwaltung von Mitarbeitern, Produkten, Bestellungen
Qualitative Ziele	Optisch ansprechende Oberfläche, Gestensteuerung
Organisatorische Ziele	Fertigstellung geplant am: 1. November 2013
Funktionale Anforderungen	Schlüssige Bedienung, Multi-User Bedienung
EDV Anforderungen	Internetverbindung, Hostingprovider
Schnittstellen	Tablet, Computer, Handy
Wartbarkeit	Das Produkt wird Wartungsfrei konzipiert
Sonstiges	Arbeit wird als Diplomarbeit geführt

Tabelle 6: Pflichtenheft Projektcharakteristika

Lieferumfang

Dokumentation	In Text und Bildern
Schulung	Persönliche Schulung
Abnahme	Durch erfüllen des Pflichtenhefts
Service / Wartung	Keine Wartung. Quellcode wird mitgeliefert.
Quellcode	Zur eventuellen Fehlerbehebung

Tabelle 7: Pflichtenheft Lieferumfang

Einflussfaktoren

Allgemein

Durch eventuelle längere Krankenstände des Teams und / oder des Betreuers kann sich die Entwicklungsdauer erheblich verzögern. Im Zeitalter digitaler Medien ist eine digitale Kommunikation zwar prinzipiell möglich, dennoch kann keine Verzögerung ausgeschlossen werden.

Funktion & Design

Es gibt mehrere Accounts, damit nicht jeder Mitarbeiter auf alle Daten zugreifen oder alle Daten ändern kann. Mit „Chef“-Account können alle Daten geändert und auf alle Daten zugegriffen werden, mit dem „Mitarbeiter“-Account kann man nicht alle Daten ändern, aber zum Beispiel neue Bestellungen eingeben oder den Dienstplan verwalten. Bei Verlust der Eingabedaten eines Benutzers wird vom System eine Funktion bereitgestellt, die neue Zugangsdaten zur Verfügung stellt.

Wenn die Anwendung gestartet wird kommt man immer auf den Wochenplan. Es gibt 4 Tabs auszuwählen: Dienstplan, Wochenplan, Produktion, Einstellungen

Farbliche Hervorhebungen:

Mitarbeiter: Die Mitarbeiter werden nach Berufsgruppen farblich hinterlegt.

Rot=Konditoren; Gelb=Hilfskräfte; Blau=Lehrlinge; Grün=Leiharbeiter

Produkte: Die Produkte werden nach Produktgruppen farblich hinterlegt.

Die Produktgruppen können im Programm neu angelegt/geändert werden, dabei können auch die Farben ausgewählt werden.

Dienstplan:

Alle Konditoren, Lehrlinge und Hilfskräfte haben standardmäßig jeden Tag Dienst. Diese sind in einer Liste nach Berufsgruppen und Priorität aufgelistet. Durch ziehen in den Papierkorb können die Mitarbeiter, die an diesem Tag nicht da sind, entfernt werden. Die Arbeitszeit einzelner Personen können auch verändert werden, falls ein Mitarbeiter später beginnt oder früher aufhört. Über einen „Leiharbeiter"-Button kommt man ins Leiharbeiter-Fenster.

Leiharbeiter:

In diesem Fenster können Leiharbeiter für einen oder mehrere Tage hinzugefügt werden mit Namen und Leihfirmen. Diese sind dann auch im Tagesplan einzuteilen.

Wochenplan:

Im Wochenplan sind für jeden Tag alle Arbeitsschritte mit Zeiten aufgelistet. Rechts gibt es eine Leiste mit allen Produkten, die unter Produktion ausgewählt wurden, die durch ziehen zu bestimmten Tagen an diesen Tagen zugeteilt werden. Aus der Leiste können auch Produkte entfernt werden, falls zu viel hinzugefügt wurden. Über einen „Heute"-Button kann man im Kalender zum aktuellen Tag springen. Für jeden Tag gibt es eine Auflistung der Anzahl an Mitarbeitern nach Berufsgruppen, dann die gesamte Zeit aller Mitarbeiter und die Zeit, die noch verfügbar(grün) / zu viel(rot) ist. Die Anzahl der benötigten Leiharbeiter wird gleich berechnet. Durch verschieben eines einzelnen Arbeitsschrittes werden alle anderen Arbeitsschritte dieses Produkts um genauso viele Tage verschoben. Es gibt auch die Möglichkeit die Produktkette zu zerreißen. Die Produktmengen können auch gleich im Wochenplan geändert werden. Durch drücken auf die einzelnen Tage kommt man zum Tagesplan.

Tagesplan:

Im Tagesplan gibt es für jeden Posten eine Auflistung der Anzahl an Mitarbeitern nach Berufsgruppen, dann die gesamte Zeit aller Mitarbeiter und die Zeit, die noch verfügbar(grün) / zu viel(rot) ist. Durch klicken auf den Balken mit den Mitarbeiteranzahlen öffnet sich ein neues Fenster, wo alle Mitarbeiter aufgelistet sind, man einzelne löschen und einen Verantwortlichen festlegen kann. Rechts gibt es eine Leiste mit allen Mitarbeitern, die an diesem Tag anwesend sind, die durch ziehen zu den Posten dort zugeteilt werden. Arbeitsschritte können zwischen Posten verschoben werden. Die Mengen der Produkte können auch noch im Tagesplan verändert werden. Über einen Button kann man für jeden Posten eine Liste mit allen Arbeitsschritten Mitarbeitern und dem Verantwortlichen ausdrucken. Über einen weiteren Button kann man den Drucker einrichten.

Produktion:

Alle Produkte sind nach Produktgruppen und Alphabet aufgelistet und durch drücken auf das jeweilige Produkt können die verfügbaren Mengen ausgewählt werden. Alle

ausgewählten Produkte sind dann im Wochenplan in der Leiste. Mit einem Button gelangt man zu den Sonderbestellungen.

Sonderbestellungen:

Hier können Sonderbestellungen(Hochzeitstorten, einmalige Bestellungen,…) eingegeben werden. Diese werden automatisch gespeichert und dann unter Sonderbestellungen aufgelistet und dann wiederverwendet/verändert. Über einen Button können neue Sonderbestellungen hinzugefügt werden. Sonderbestellungen werden so eingegeben wie unten unter Einstellungen beschrieben.

Einstellungen

Die Einstellungen können nur mit dem „Chef“-Account geändert werden.

Unter Einstellungen gibt es wieder 4 Tabs auszuwählen: Produkte, Mitarbeiter, Lieferanten, Posten

Produkte

Die Produkte sind nach Produktgruppen und Alphabet aufgelistet und können verändert/gelöscht werden. Über einen Button können neue Produkte hinzugefügt werden. Parameter für ein Produkt sind der Name, Auswahl der Produktgruppe und die Arbeitsschritte. Über einen Button können weitere Arbeitsschritte hinzugefügt werden. Parameter für einen Arbeitsschritt sind der Name, Auswahl vom Posten, Mengen und dazugehörige Zeiten, Zutaten für diesen Arbeitsschritt, der Tag in der Produktkette. Einzelne Produkte können kopiert und modifiziert werden.

Mitarbeiter

Die Mitarbeiter sind nach Produktgruppen und Alphabet aufgelistet und können verändert/gelöscht werden. Über einen Button können neue Mitarbeiter hinzugefügt werden. Parameter für einen Mitarbeiter sind ein Foto, Vor- und Nachname, Auswahl der Berufsgruppe und das Geburtsdatum.

Lieferanten

Die Lieferanten sind nach Alphabet aufgelistet und können verändert/gelöscht werden. Über einen Button können neue Lieferanten hinzugefügt werden. Parameter für einen Lieferanten sind der Name, eine Auswahlmöglichkeit(wöchentlich, 2-wöchentlich, 3-wöchentlich, 4-wöchentlich) und die einzelnen Wochentage, an denen beim Lieferanten bestellt werden muss, müssen auch ausgewählt werden. Feiertage werden automatisch berücksichtigt. Die Lieferantenbestellungen werden automatisch beim Posten Firmenleitung aufgelistet.

Posten

Die vorhandenen Posten sind nach Alphabet aufgelistet und können verändert/gelöscht werden. Über einen Button können neue Posten hinzugefügt werden.

Risiken

Beurteilung

Geringe Risiken	Verzögerung der Fertigstellung
Mittlere Risiken	Geschmack des Kunden verfehlt, Bedienkonzept nicht realisierbar bzw. verfehlt
Hohe Risiken	Unternehmerisches Risiko

Tabelle 8: Pflichtenheft Risiken

Verpflichtungen

Finanzielles

Für alle oben genannten Leistungen ist Heiss & Süß GmbH verpflichtet die Herren Thoma und Weise (Auftragnehmer) jeweils mit einem Nettobetrag von 1.000,- Euro zu entlohnen. Da dies unter die Einkommenssteueruntergrenze fällt werden voraussichtlich keine Steuern anfallen.

II. Werkvertrag

Dieser Vertrag wird zwischen der Firma ***Heiss & Süß GmbH***(Auftraggeber) und den Herren ***Martin Thoma*** und ***Martin Weise***(Auftragnehmer) abgeschlossen und beinhaltet folgende Aspekte:

1. **Arbeitsverhältnis**

   Das Arbeitsverhältnis beginnt mit Vertragsabschluss und endet mit Erfüllung der vorgesehenen Tätigkeit. Der späteste Zeitpunkt, da das Werk im Rahmen einer Diplomarbeit erstellt wird, ist der 1. November 2013.

2. **Vorgesehene Tätigkeit**

   Die Vertragspartner ***Thoma*** und ***Weise*** sind verpflichtet eine Webanwendung für ***Heiss & Süß GmbH*** zu entwerfen und umzusetzen, die dessen Vorgaben entspricht (im Pflichtenheft geregelt). Der Umfang der vorgesehenen Tätigkeit basiert auf Grundlage des Pflichtenhefts. Alle darüber hinausgehenden Tätigkeiten sind nicht Teil des Vertrags.

3. **Arbeitsort & Arbeitszeit**

   Die Auftragnehmer unterliegen hinsichtlich Arbeitszeit, Arbeitsort und konkreter Durchführung der Tätigkeit keinen Weisungen des Auftraggebers.

4. **Haftung**

   Der Auftraggeber trägt das Risiko, dass er die vom Auftragnehmer geleistete Arbeit nutzbringend einsetzen kann. Ob die erbrachte Arbeit dem Auftraggeber einen finanziellen Nutzen bringt, spielt keine Rolle. Die Abwälzung des unternehmerischen Risikos auf die

Auftragnehmer ist unzulässig. Die Auftragnehmer haften jedoch dafür, dass der Auftrag entsprechend dem Pflichtenheft ordnungsgemäß und zum vereinbarten Termin erbracht wird.

5. **Rechte**

Urheber der geschaffenen webtechnischen Anwendung ist die Heiss & Süß GmbH und diese kann vollständig über das Werk verfügen. Sämtliche Rechte (wie Patent-, Urheber- und Leistungsschutzrechte) bleiben derFirma Heiss & Süß vorbehalten. Das Designrecht bleibt den Herren Thoma und Weise vorbehalten.

Im Falle des Verkaufs der webtechnischen Anwendung bekommen 30% des Nettoumsatzes die Herren Thoma und Weise. Und des Weiteren werden die Herren Thoma und Weise als Support empfohlen.

6. **Ansprüche**

Der Auftraggeber verpflichtet sich die Auftragnehmer zu entlohnen, wenn diese Ihre Arbeit zur Verfügung stellen. Diese Entlohnung beträgt insgesamt Euro 2.000,- ohne MwSt., zahlbar prompt nach Rechnungserhalt.

III. Ablaufdiagramme

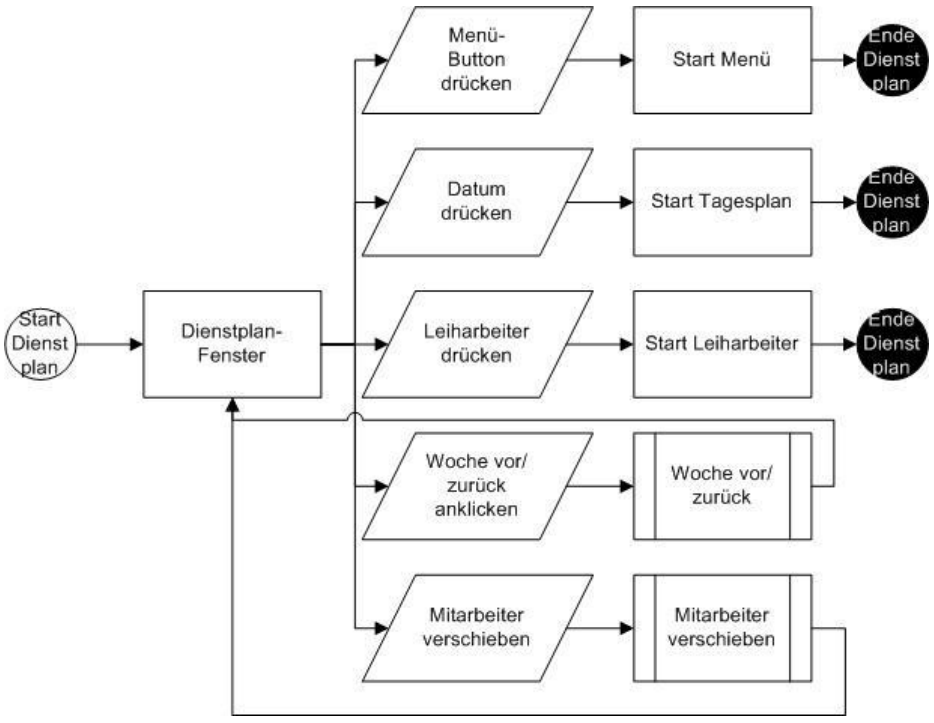

Abb. 41: Flussdiagramm Dienstplan

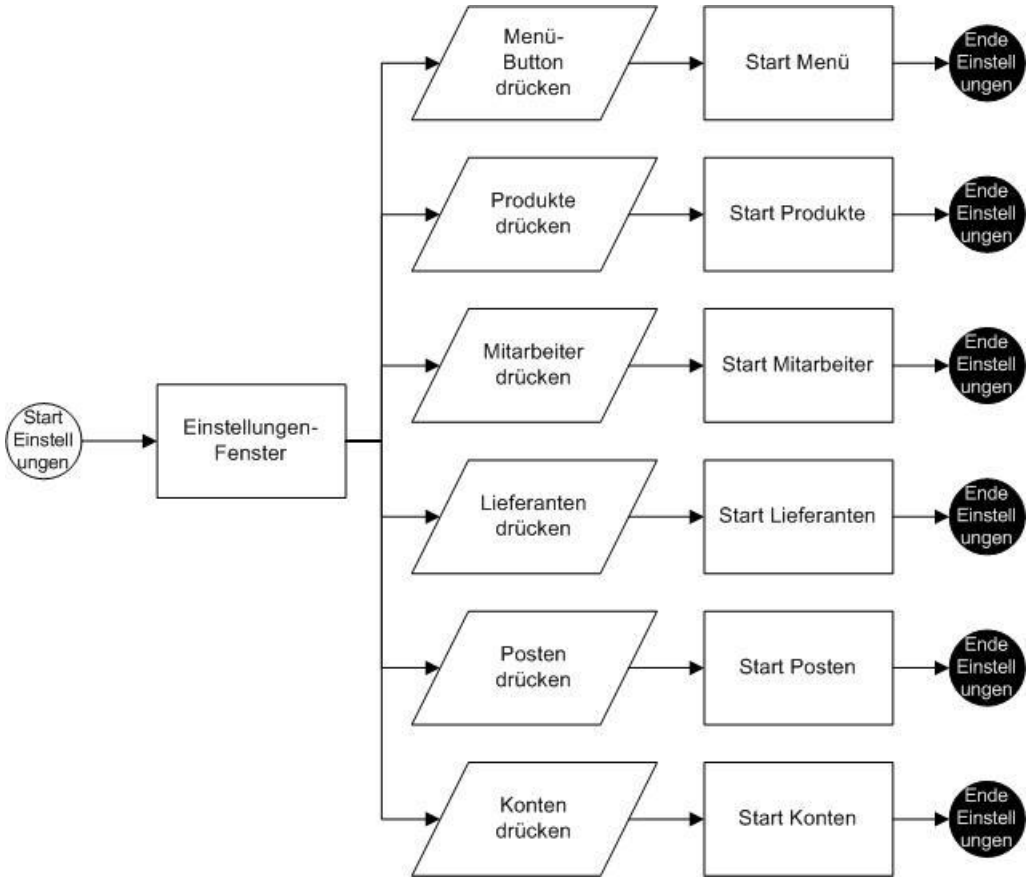

Abb. 42: Flussdiagramm Einstellungen

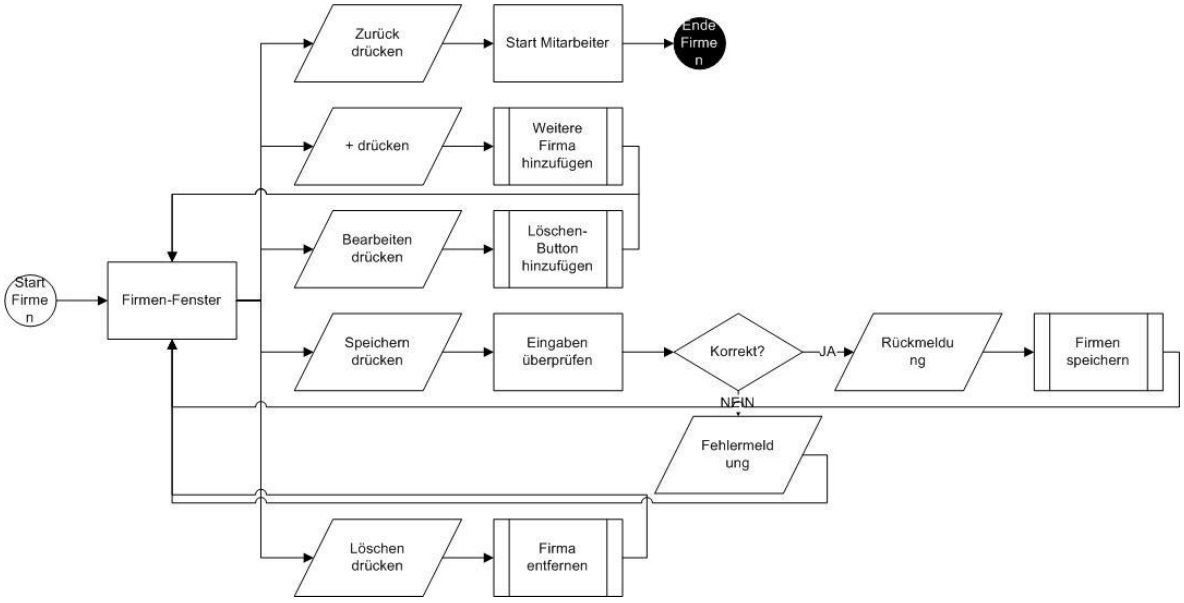

Abb. 43: Flussdiagramm Firmen

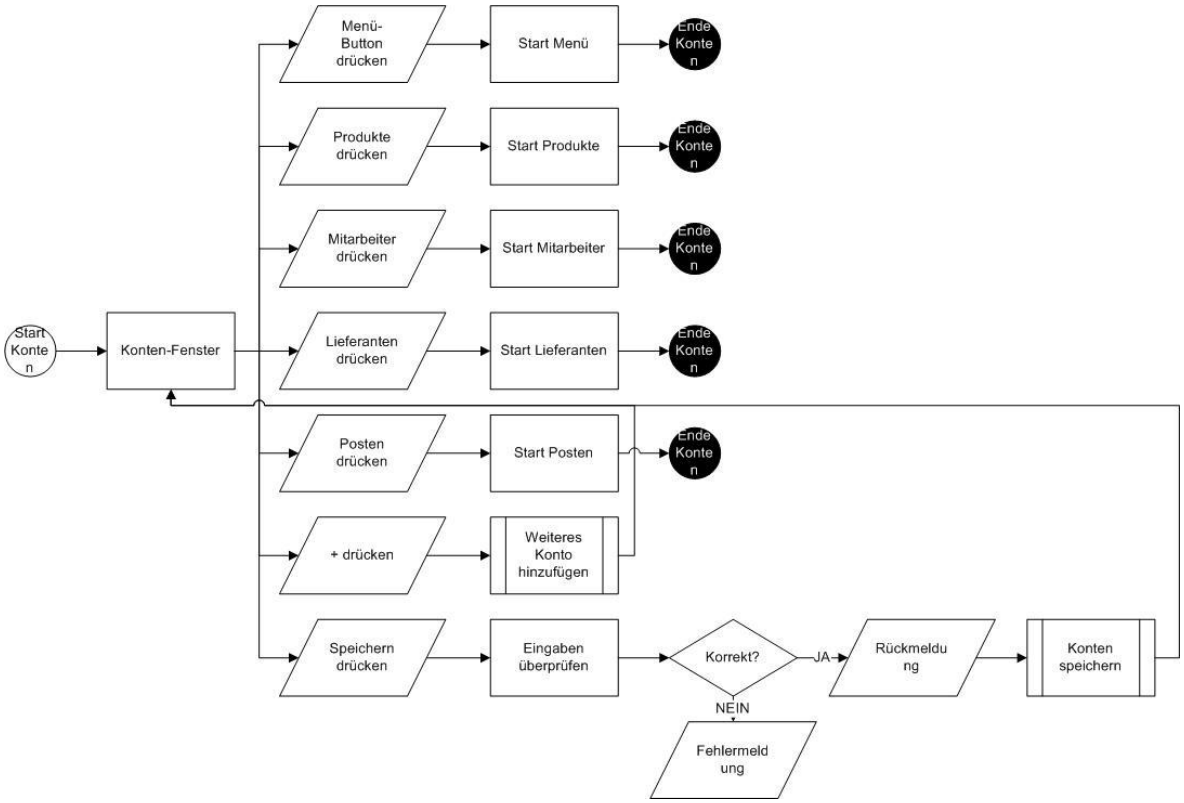

Abb. 44: Flussdiagramm Konten

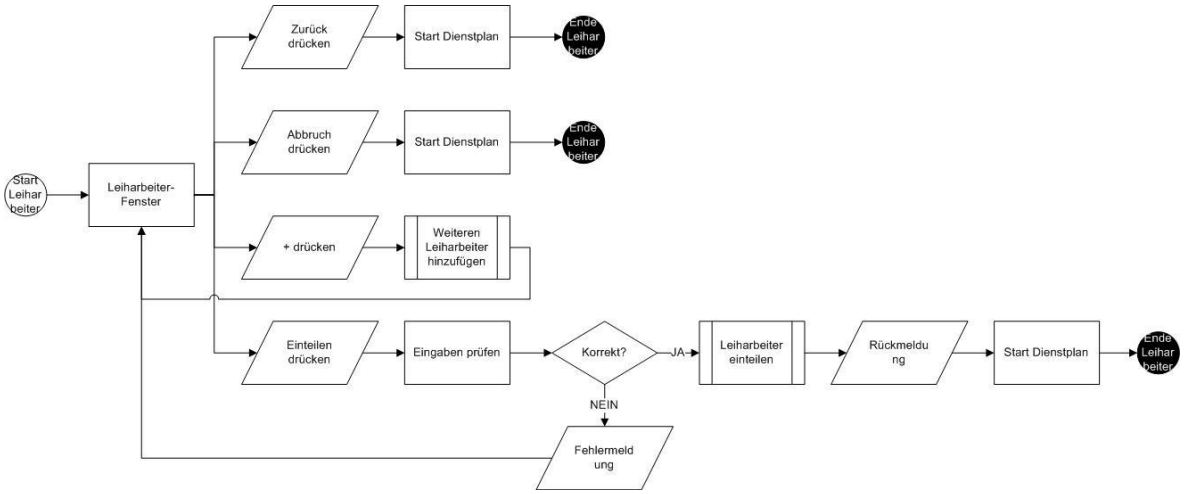

Abb. 45: Flussdiagramm Leiharbeiter

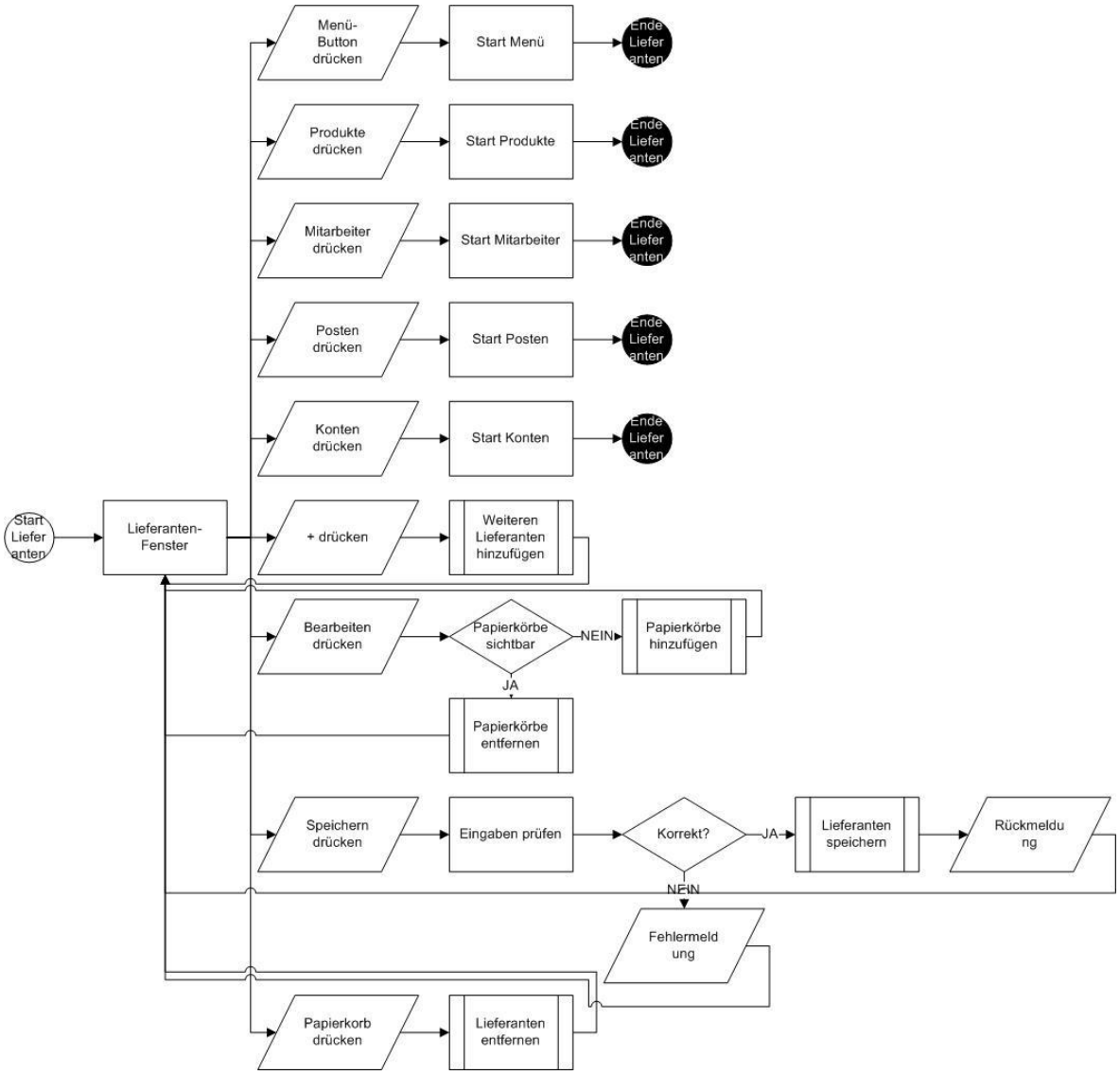

Abb. 46: Flussdiagramm Lieferanten

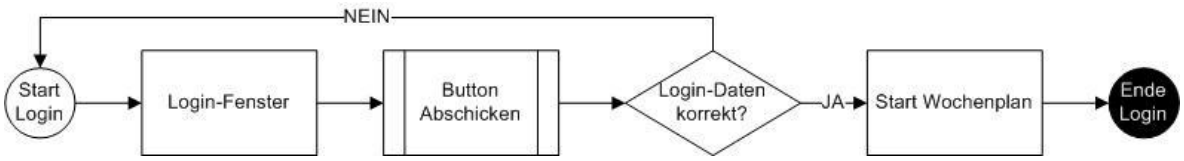

Abb. 47: Flussdiagramm Login

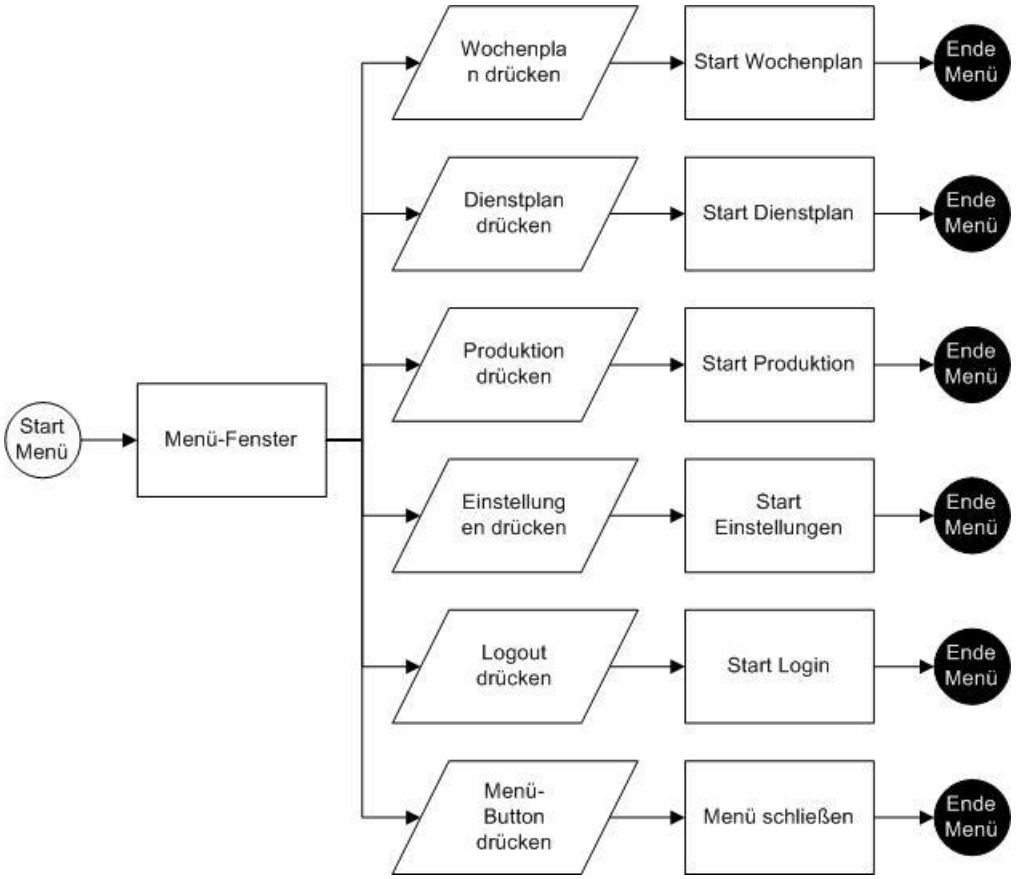

Abb. 48: Flussdiagramm Menü

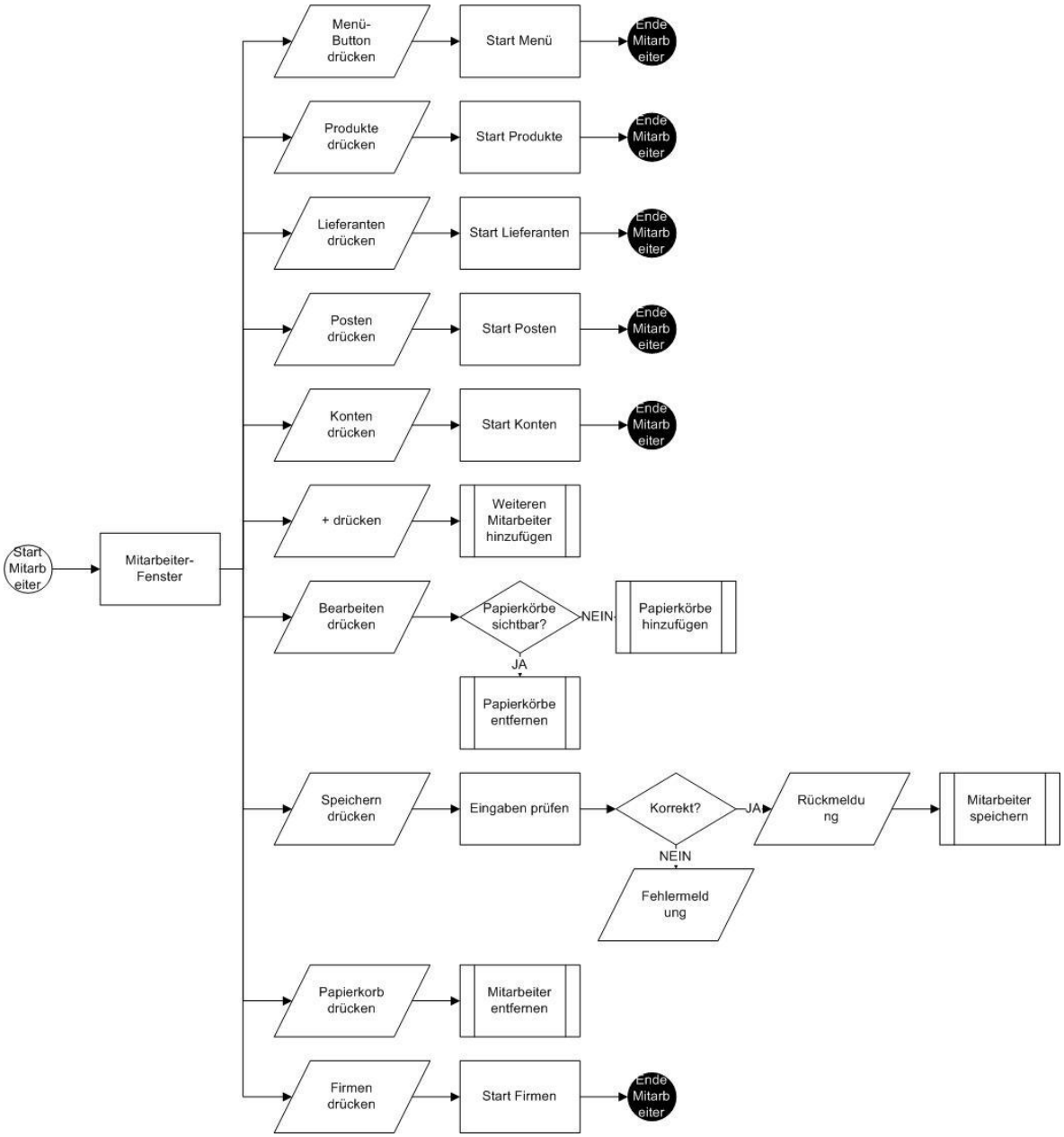

Abb. 49: Flussdiagramm Mitarbeiter

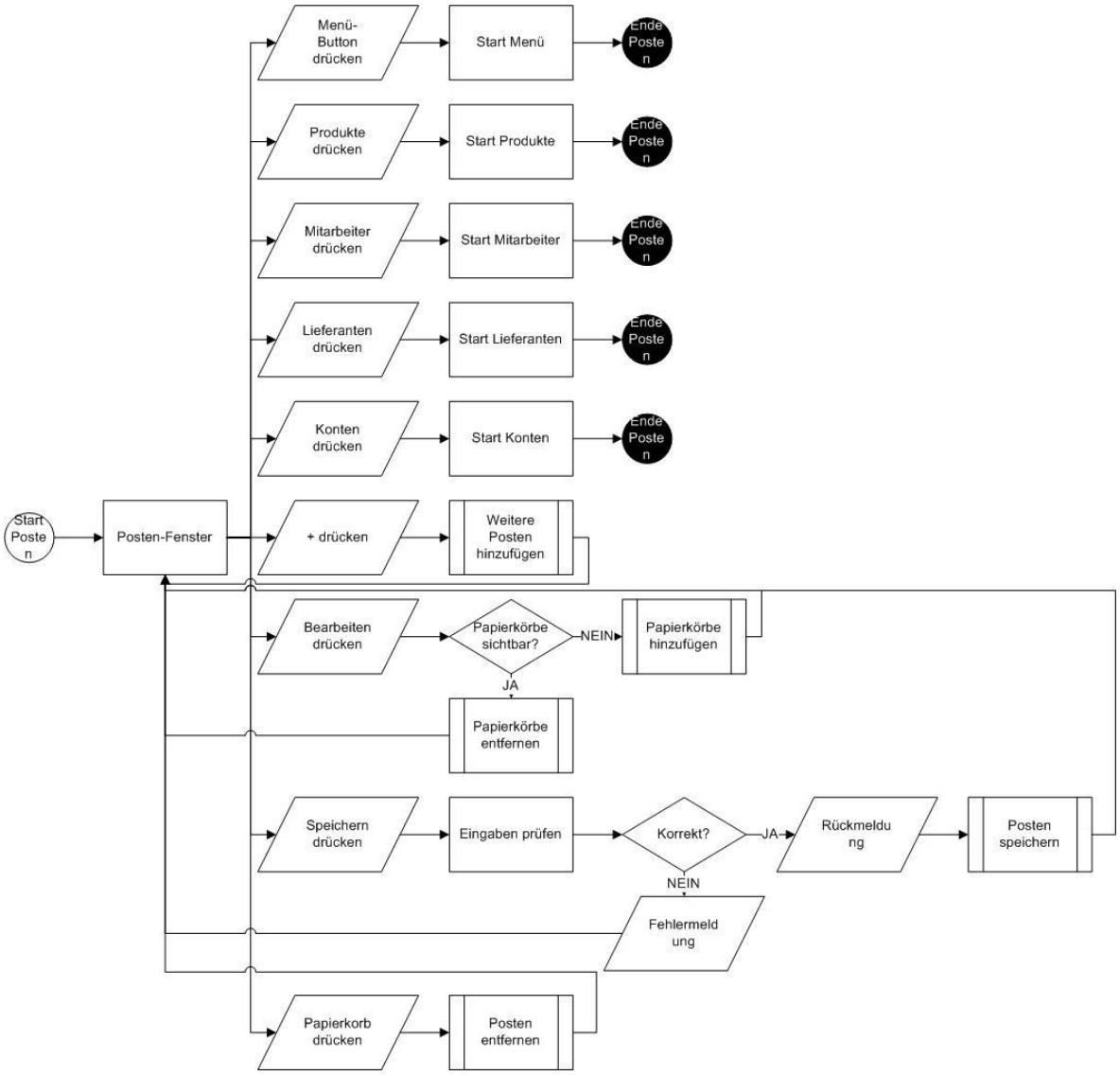

Abb. 50: Flussdiagramme Posten

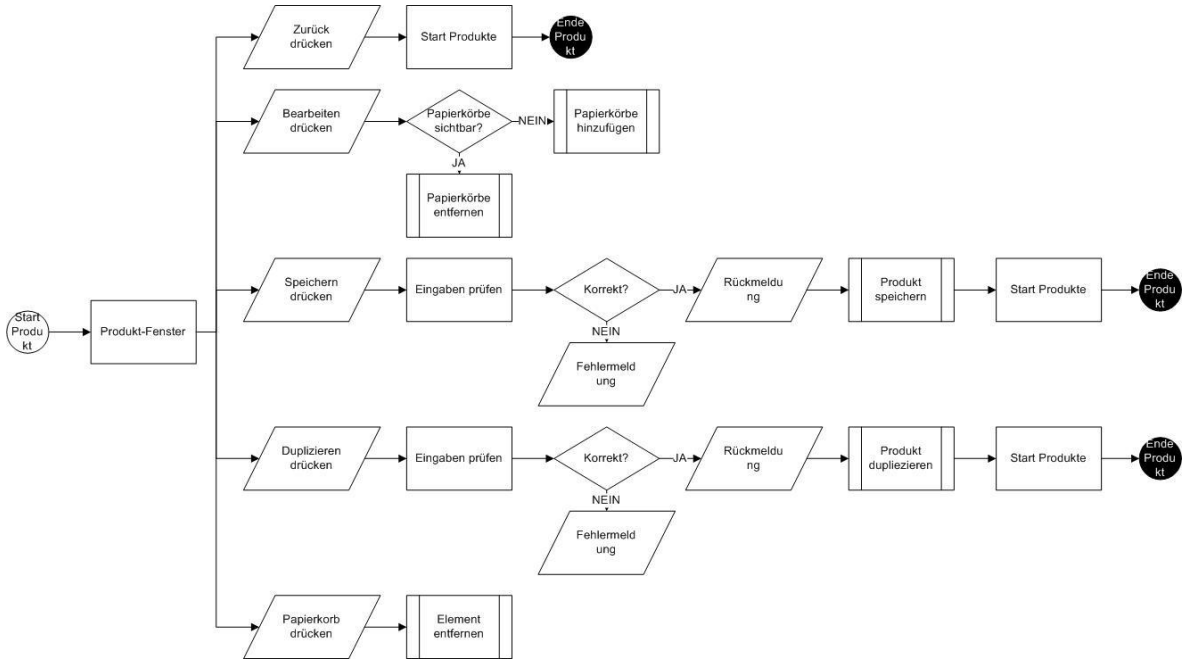

Abb. 51: Flussdiagramm Produkt

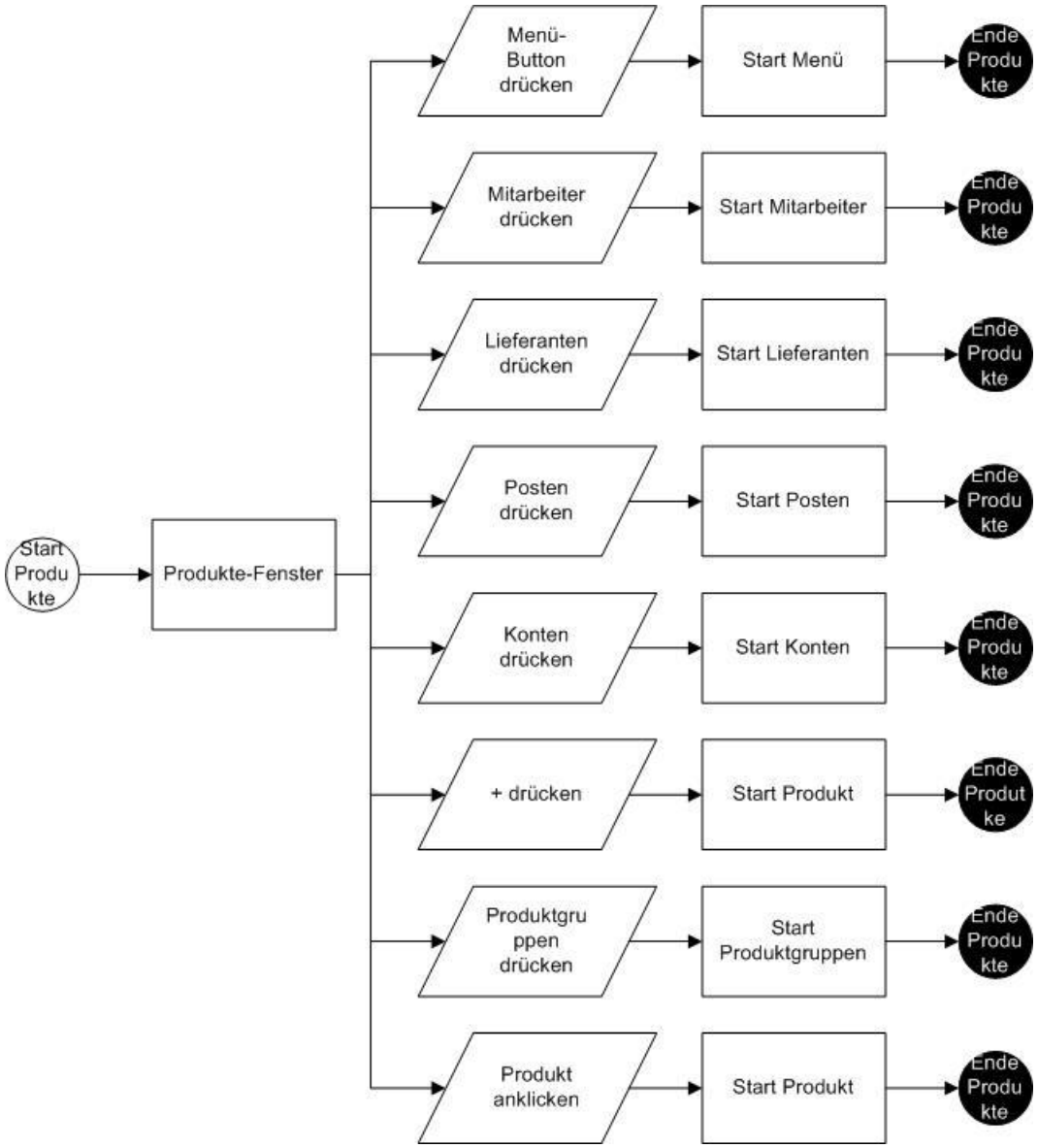

Abb. 52: Flussdiagramm Produkte

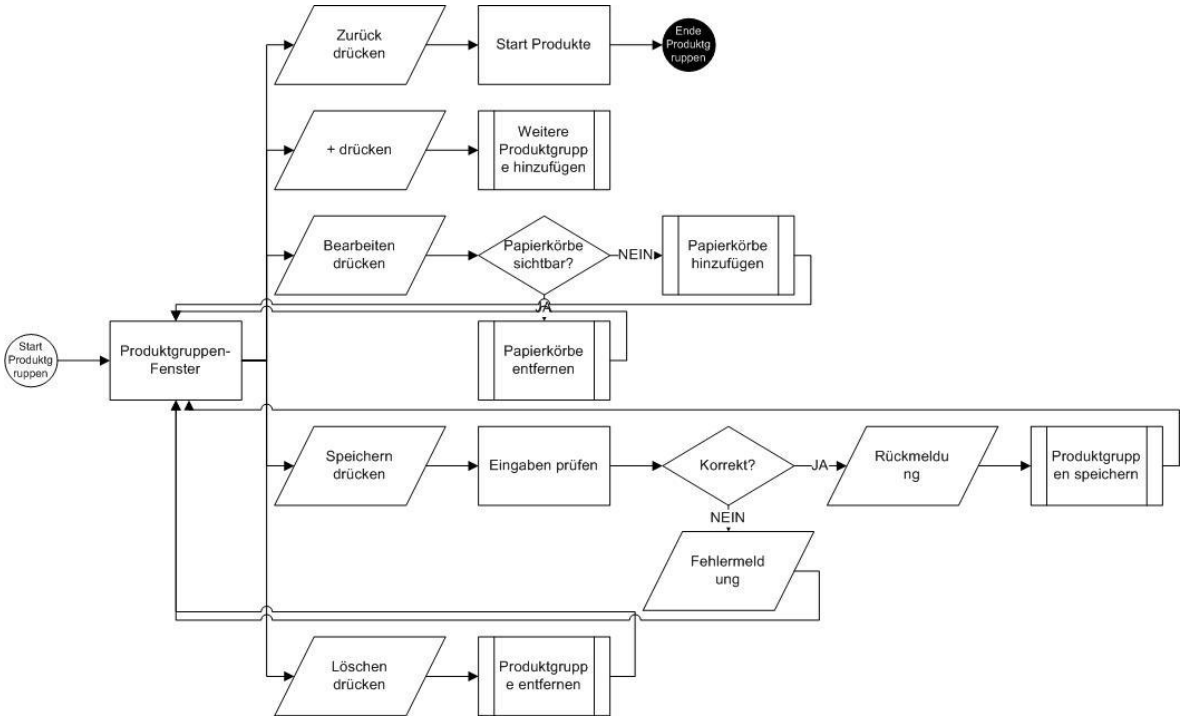

Abb. 53: Flussdiagramm Produktgruppen

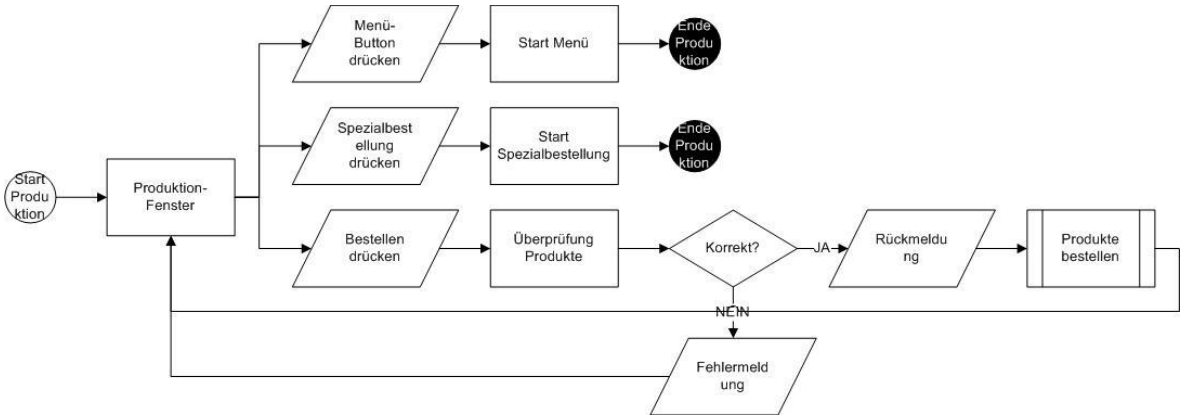

Abb. 54: Flussdiagramm Produktion

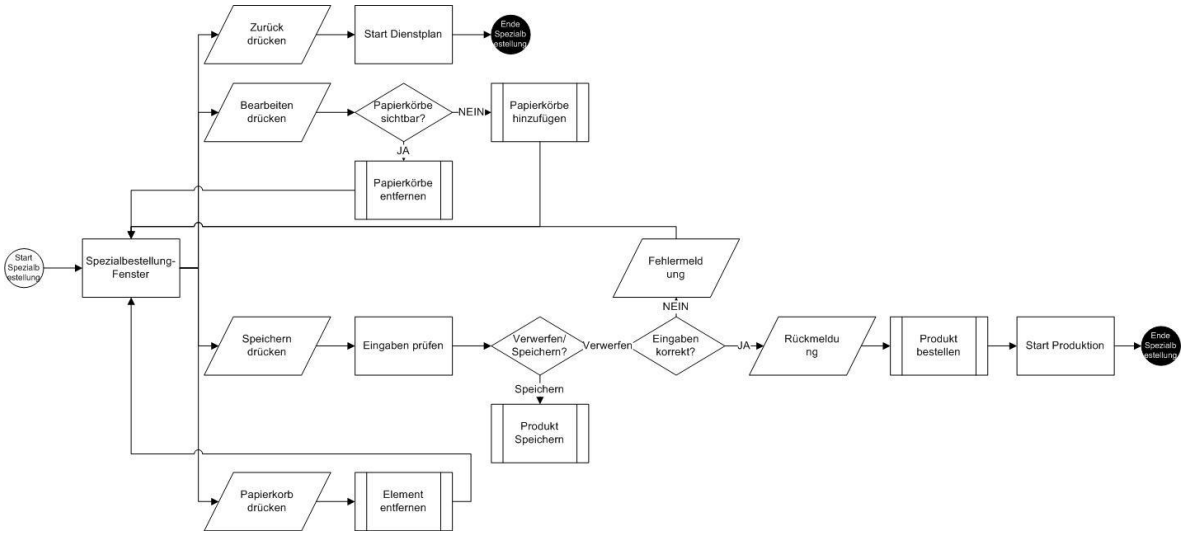

Abb. 55: Flussdiagramm Spezialbestellung

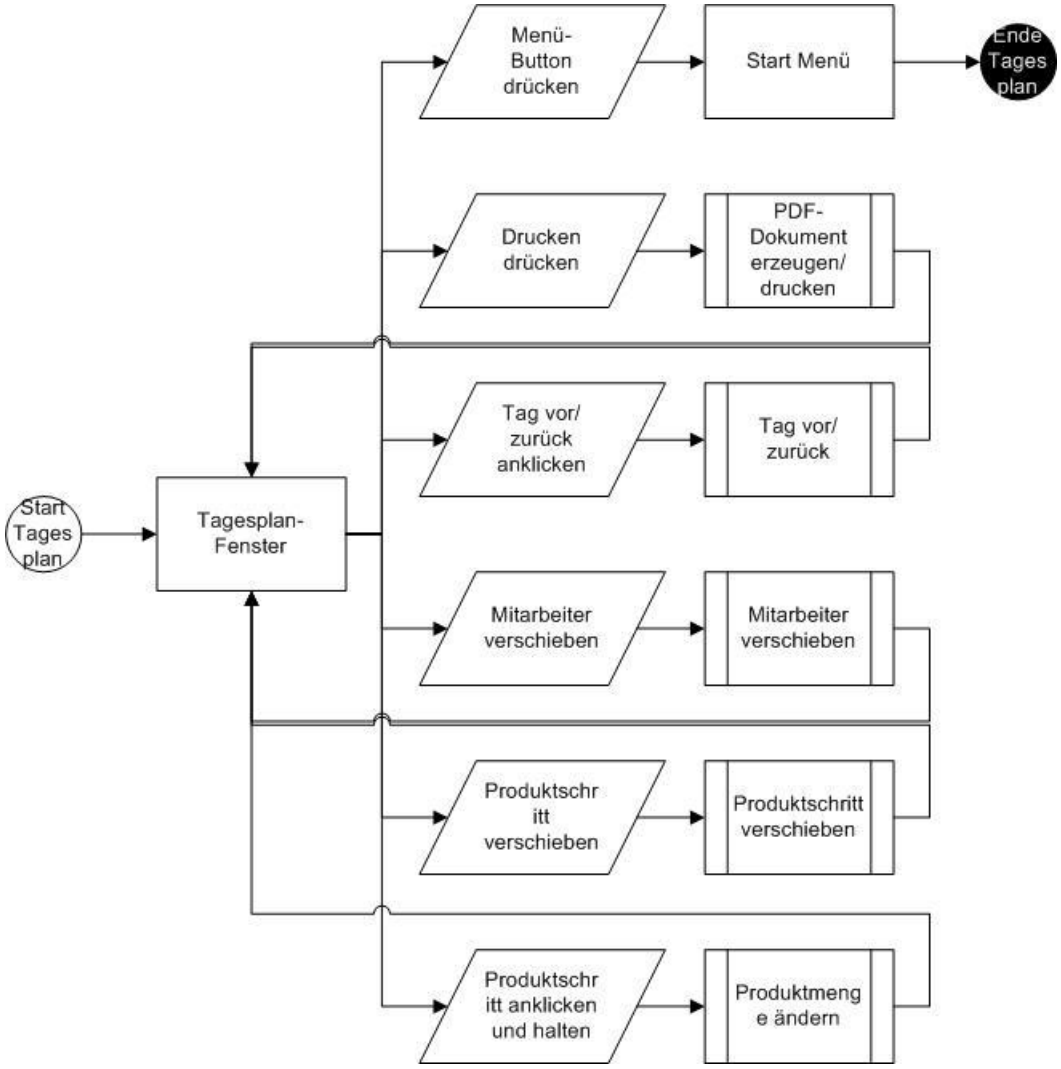

Abb. 56: Flussdiagramm Tagesplan

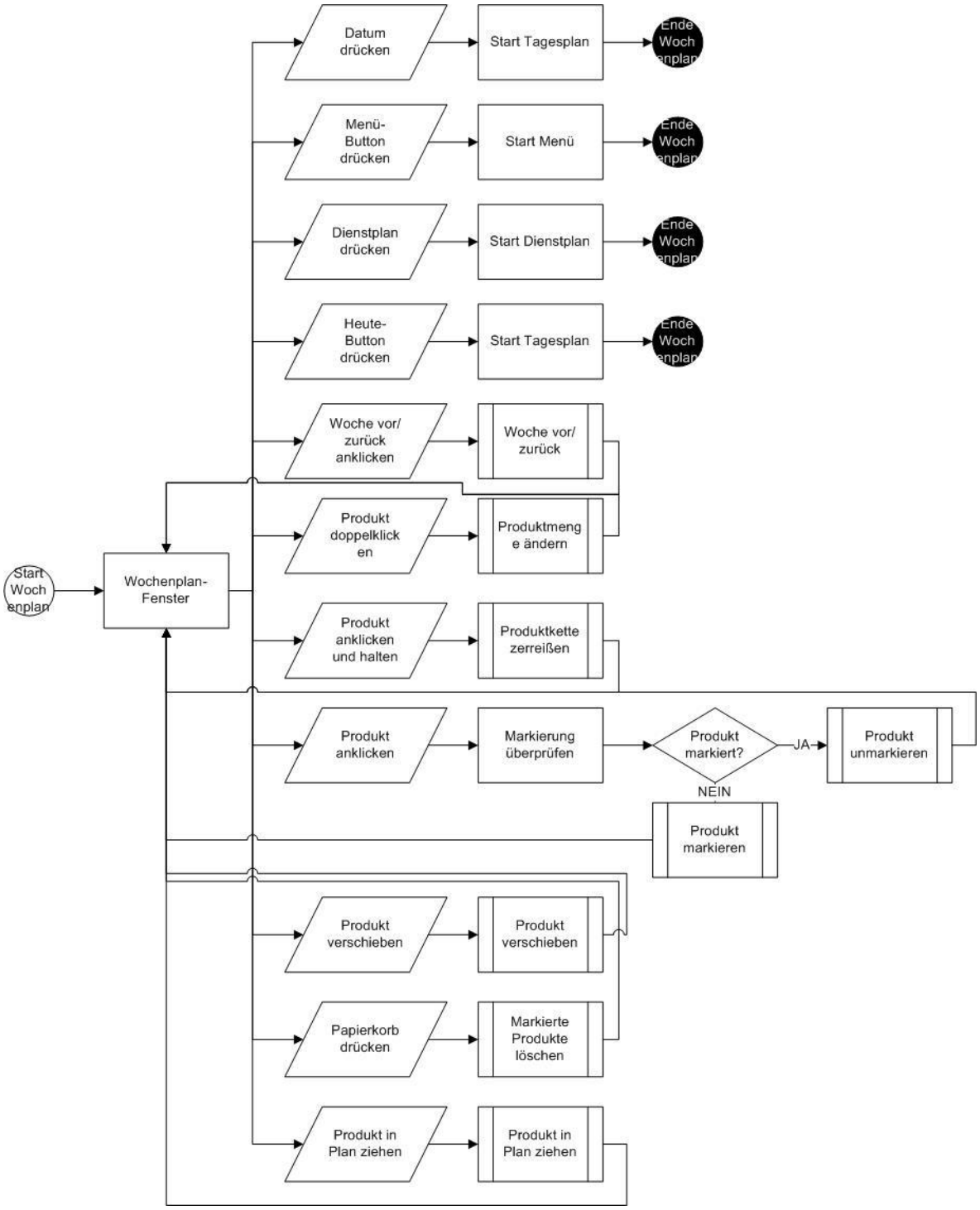

Abb. 57: Flussdiagramm Wochenplan

IV. Zeiterfassung

Name	Datum	Tätigkeit	Dauer (h)	Bereich
Thoma	08.02.2013	Diplomarbeitsvorschlag von Herrn Tutschek	2	P
Weise	08.02.2013	Diplomarbeitsvorschlag von Herrn Tutschek	2	P
Thoma	15.02.2013	Diplomarbeitsvorschlag konkretisieren	2	P
Weise	15.02.2013	Diplomarbeitsvorschlag konkretisieren	2	P
Thoma	21.02.2013	Kundentreffen (Anforderungen)	4	P
Weise	21.02.2013	Kundentreffen (Anforderungen)	4	P
Thoma	01.04.2013	Lastenheft erstellen	5	P
Weise	01.04.2013	Lastenheft erstellen	5	P
Thoma	09.04.2013	Pflichtenheft erstellen	3	P
Weise	09.04.2013	Pflichtenheft erstellen	3	P
Thoma	23.04.2013	Pflichtenheft erstellen	4	P
Weise	23.04.2013	Pflichtenheft erstellen	4	P
Thoma	30.04.2013	Pflichtenheft erstellen	3	P
Weise	30.04.2013	Pflichtenheft erstellen	3	P
Thoma	06.03.2013	Designentwurf lt. Pflichtenheft	4	S
Weise	06.03.2013	Designentwurf lt. Pflichtenheft	4	S
Thoma	19.03.2013	Designentwurf lt. Pflichtenheft	5	S
Weise	19.03.2013	Designentwurf lt. Pflichtenheft	5	S
Thoma	02.04.2013	Design/Konzept-Präsentation	2	P
Weise	02.04.2013	Design/Konzept-Präsentation	2	P
Thoma	03.04.2013	Pflichtenheft verbessern	4	P
Weise	03.04.2013	Pflichtenheft verbessern	4	P
Thoma	11.05.2013	Pflichtenheft verbessern	5	P
Weise	11.05.2013	Pflichtenheft verbessern	5	P
Thoma	16.05.2013	Kundentreffen (Pflichtenheft unterzeichnen)	2	P
Weise	16.05.2013	Kundentreffen (Pflichtenheft unterzeichnen)	2	P
Thoma	17.05.2013	Wochenplan, Dienstplan Layout #1	4	P
Weise	17.05.2013	Wochenplan, Dienstplan Layout #1	4	P
Weise	12.06.2013	Überall App Congress	6	X
Weise	13.06.2013	Überall App Congress	6	X
Thoma	17.06.2013	Verwerfen Layout #1, Erstellen Layout #2	4	P
Weise	17.06.2013	Verwerfen Layout #1, Erstellen Layout #2	4	P
Thoma	25.06.2013	Frontend Arbeiten	4	S
Weise	25.06.2013	Frontend Arbeiten	4	S
Thoma	28.06.2013	Frontend Arbeiten	8	S
Thoma	01.07.2013	Frontend Arbeiten	4	S
Weise	01.07.2013	Frontend Arbeiten	4	S
Weise	10.07.2013	Backend Arbeiten	4	S
Weise	12.07.2013	Backend Arbeiten	4	S

Weise	14.07.2013	Backend Arbeiten	4	S
Weise	16.07.2013	Frontend Arbeiten	4	S
Thoma	16.07.2013	Frontend Arbeiten	5	S
Weise	18.07.2013	Backend Arbeiten	3	S
Weise	19.07.2013	Backend Arbeiten	3	S
Thoma	20.07.2013	Frontend Arbeiten	4	S
Thoma	21.07.2013	Frontend Arbeiten	4	S
Weise	22.07.2013	Backend Arbeiten	4	S
Weise	01.08.2013	Backend Arbeiten	2	S
Weise	02.08.2013	Backend Arbeiten	2	S
Weise	03.08.2013	Backend Arbeiten	2	S
Weise	05.08.2013	Backend Arbeiten	2	S
Weise	07.08.2013	Backend Arbeiten	2	S
Weise	08.08.2013	Backend Arbeiten	2	S
Thoma	18.08.2013	Kundentreffen / Entwurfpräsentation	5	P
Weise	18.08.2013	Kundentreffen / Entwurfpräsentation	5	P
Thoma	27.08.2013	Sektor5 Treffen	6	X
Weise	27.08.2013	Sektor5 Treffen	6	X
Thoma	28.08.2013	Frontend Arbeiten	4	S
Weise	28.08.2013	Frontend Arbeiten	4	S
Thoma	01.09.2013	Frontend Arbeiten	4	S
Thoma	02.09.2013	Frontend Arbeiten	2	S
Thoma	03.09.2013	Frontend Arbeiten	4	S
Weise	05.09.2013	Backend Arbeiten	2	S
Weise	06.09.2013	Backend Arbeiten	2	S
Weise	07.09.2013	Backend Arbeiten	2	S
Thoma	23.09.2013	Diplomarbeitsantrag	2	S
Weise	23.09.2013	Diplomarbeitsantrag	2	S
Thoma	30.09.2013	SQL-Abfragen, Programmierung	6	S
Weise	30.09.2013	SQL-Abfragen, Programmierung	6	S
Thoma	07.10.2013	Bedienungsanleitung	6	D
Weise	07.10.2013	Programmierung	6	S
Thoma	13.10.2013	Kundentreffen	2	P
Weise	13.10.2013	Kundentreffen	2	P
Thoma	14.10.2013	Bedienungsanleitung	6	D
Weise	14.10.2013	Programmierung	6	S
Thoma	21.10.2013	Bedienungsanleitung	4	D
Thoma	21.10.2013	Datenbank-Optimierung	4	S
Weise	28.10.2013	Datenbank-Optimierung	4	S
Thoma	11.11.2013	Vorbereitung vorläufige Projektübergabe	4	S
Weise	11.11.2013	Vorbereitung vorläufige Projektübergabe	4	S
Thoma	17.11.2013	Kundentreffen Fehler beheben	2	S

Weise	17.11.2013	Kundentreffen Fehler beheben	2	S
Thoma	17.11.2013	Kundentreffen (vorl. Projektübergabe f. Testen)	0,5	P
Weise	17.11.2013	Kundentreffen (vorl. Projektübergabe f. Testen)	0,5	P
Thoma	09.12.2013	Vorbereitung endgültige Projektübergabe	6	S
Weise	09.12.2013	Vorbereitung endgültige Projektübergabe	6	S
Thoma	22.12.2013	Softwareübergabe	2	S
Weise	22.12.2013	Softwareübergabe	2	S
Thoma	27.12.2013	Rechnung gestellt	1	P
Weise	27.12.2013	Rechnung gestellt	1	P
Thoma	06.01.2013	Jugend Innovativ Unterlagen Zusammenstellen	8	X
Weise	06.01.2013	Jugend Innovativ Unterlagen Zusammenstellen	4	X
Weise	03.02.2013	Beginn Diplomarbeit	2	D
Thoma	10.02.2014	Projektabschluss	2	S
Weise	10.02.2014	Projektabschluss	2	S
Weise	12.02.2014	Theorie	4	D
Weise	14.02.2014	Theorie	4	D
Thoma	14.02.2013	Diplomarbeit	5	D
Weise	15.02.2014	Theorie	4	D
Weise	17.02.2014	Theorie	4	D
Thoma	18.02.2014	Diplomarbeit	3	D
Weise	20.02.2014	Theorie	4	D
Weise	21.02.2014	Theorie	4	D
Weise	21.02.2014	Durchführung	4	D
Weise	22.02.2014	Durchführung	4	D
Thoma	01.03.2014	Kod.io Developer Conference App Pitch	6	X
Weise	01.03.2014	Kod.io Developer Conference App Pitch	6	X
Weise	03.03.2014	Durchführung	1	D
Weise	05.03.2014	Durchführung	2	D
Thoma	27.03.2014	Accent Präsentation	3	X
Weise	27.03.2014	Accent Präsentation	3	X
Weise	30.03.2014	Durchführung	4	D
Weise	07.04.2014	Zeitaufstellung	6	D
Thoma	14.04.2014	Diplomarbeit	4	D
Weise	14.04.2014	Diplomarbeit	4	D
Thoma	15.04.2014	Diplomarbeit	3	D
Thoma	16.04.2014	Diplomarbeit	5	D
Thoma	17.04.2014	Diplomarbeit	4	D
Thoma	19.04.2014	Diplomarbeit	4	D
Thoma	21.04.2014	Diplomarbeit	5	D
Thoma	22.04.2014	Diplomarbeit	3	D
Weise	28.04.2014	Diplomarbeit	1	D

Tabelle 9: Zeiterfassung Detailaufschlüsselung

V. DVD

Inhalt:

- Diplomarbeit im Format .pdf
- Diplomarbeit im Format .docx
- Zusammenfassung der Diplomarbeit (DE) im Format .docx
- Zusammenfassung der Diplomarbeit (EN) im Format .docx
- Präsentation der Diplomarbeit im Format .pptx
- Source-Code in einem komprimierten Verzeichnis im Format .rar